



**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Εφαρμογές Memristors Σε VLSI Κυκλώματα**

**Βασίλειος Κ. Αθανασόπουλος**

**Επιβλέπουσα: Αγγελική Αραπογιάννη**

**ΑΘΗΝΑ**

**ΙΟΥΝΙΟΣ 2016**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Εφαρμογές Memristors Σε VLSI Κυκλώματα**

**Βασίλειος Κ. Αθανασόπουλος**

**A.M.: 1115200700306**

**Επιβλέπουσα: Αγγελική Αραπογιάννη**

## ΠΕΡΙΛΗΨΗ

Οι βελτιώσεις στις επιδόσεις των υπολογιστών τις τελευταίες δεκαετίες έχουν συνδεθεί άρρηκτα με την σμίκρυνση της CMOS τεχνολογίας, ενώ η βασική δομή των ψηφιακών υπολογιστικών συστημάτων έχει βασιστεί στην αρχιτεκτονική Von Neumann, όπου η μνήμη και η μονάδα επεξεργασίας είναι διαχωρισμένες, χρησιμοποιώντας διάφορους τύπους διασύνδεσης για την μεταξύ τους επικοινωνία. Ωστόσο οι συμβατικές τεχνολογίες μνήμης όπως της Flash, DRAM, και SRAM, φέρεται να είναι ανήκανες να συνεχίσουν αυτήν την κούρσα βελτιώσεων. Η Flash μνήμη για παράδειγμα έχει ήδη φτάσει στα φυσικά της όρια και δεν μπορεί να σμικρυνθεί περαιτέρω, κυρίως λόγω της οριακής της αντοχής. Τα προβλήματα που αντιμετωπίζουν αυτές οι τεχνολογίες μνήμης φέρεται να βρίσκουν λύση με νέες ημιαγωγικές συσκευές γνωστές και ως memristors.

Τα memristors είναι συσκευές δύο τερματικών με διαφέρουσα αντίσταση, όπου η συμπεριφορά τους είναι ανάλογη της ιστορίας τους. Τα τελευταία χρόνια, πολλά φυσικά φαινόμενα της μεταγωγής αντίστασης έχουν συνδεθεί με την θεωρητική έννοια ενός memristor, και πολλές ανερχόμενες συσκευές μνήμης τώρα πια θεωρούνται memristors. Σημαντικά χαρακτηριστικά των memristors εκτός των άλλων είναι και η υψηλή ταχύτητα, η χαμηλή κατανάλωση ενέργειας η συμβατότητα με τη CMOS τεχνολογία. Αυτή η εργασία παρουσιάζει μια προσπάθεια καταγραφής και ανάλυσης των εφαρμογών των memristors σε VLSI κυκλώματα. Στο πρώτο μέρος γίνεται λόγος για τις νέες τεχνολογίες μνήμης οι οποίες υιοθετούν την τεχνολογία των memristors και οι οποίες θεωρείται πως στο άμεσο μέλλον θα είναι οι κυρίαρχες και πιο ευρέως διαδεδομένες για την κατασκευή υπολογιστών. Ακόμα εκτός από τις τεχνολογίες μνήμης γίνεται και εκτενής αναφορά στις νέες λογικές οικογένειες αλλά και αρχιτεκτονικές οι οποίες αναπτύχθηκαν με βάση είτε τα memristors είτε συνδυασμού memristors με την συμβατική τεχνολογία CMOS.

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ:** VLSI Κυκλώματα

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:** memristor, memristive συσκευή, μνήμη μόνιμης αποθήκευσης, λογική οικογένεια, λογική πύλη

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΠΡΟΛΟΓΟΣ.....</b>	<b>9</b>
<b>1. ΕΙΣΑΓΩΓΗ ΣΤΑ MEMRISTORS.....</b>	<b>10</b>
1.1 Γενικά.....	10
1.2 Memristors.....	10
1.3 Η θεωρία των memristors.....	11
<b>2. ΕΦΑΡΜΟΓΕΣ MEMRISTORS ΣΕ ΜΝΗΜΕΣ.....</b>	<b>15</b>
2.1 Μνήμη MCAM.....	15
2.2 Resistive RAM.....	18
2.3 Σύγκριση RRAM με 3D NAND flash.....	20
<b>3. ΧΡΗΣΗ MEMRISTORS ΣΕ ΛΟΓΙΚΕΣ ΠΥΛΕΣ.....</b>	<b>24</b>
<b>3.1 Λογική υλικού συμπερασμού βασισμένη σε memristors (IMPLY).....</b>	<b>24</b>
3.1.1 Βασική λειτουργία της IMPLY λογικής πύλης.....	24
3.1.2 Ανάλυση της συμπεριφοράς μιας λογικής πύλης.....	25
3.1.3 Ταχύτητα της IMPLY λογικής.....	27
3.1.4 Εκτεταμένες λογικές λειτουργίες βασισμένες στην IMPLY.....	28
3.1.5 IMPLY μέσα σε memristor-based crossbar.....	30
3.1.6 Memristors-based crossbar.....	31
3.1.7 Μεθοδολογία σχεδιασμού λογικών πυλών.....	34
3.1.8 Παράδειγμα IMPLY λογικής πύλης ενός bit.....	40
3.1.9 Παραλλαγές στην $V_{SET}$ και $V_{COND}$ .....	42
3.1.10 Πλήρης αθροιστής οχτώ bit.....	45
3.1.11 Γενικές Boolean λειτουργίες.....	45
3.1.12 CMOS πλήρης αθροιστής.....	46
3.1.13 IMPLY πλήρης αθροιστής.....	47
3.1.14 Πέρα από τον Von Neumann – λογική μέσα στη μνήμη.....	48
3.1.15 Συμπεράσματα για IMPLY.....	50
<b>3.2 MAGIC - Memristor Aided Logic.....</b>	<b>50</b>
3.2.1 Λειτουργική αρχή της Magic.....	50
3.2.2 Magic Nor Πύλη.....	51
3.2.3 Magic μέσα σε crossbar.....	53
3.2.4 Αξιολόγηση και σχεδιαστικές εκτιμήσεις για μία Magic NOR πύλη.....	54
3.2.5 Επιπρόσθετες Magic πύλες.....	55
3.2.6 Συμπεράσματα Magic λογικής.....	56
<b>3.3 Υβριδική Memristor-CMOS λογική (MeMOS).....</b>	<b>59</b>
3.3.1 Λογική MeMOS.....	60
3.3.2 Σχεδιασμός λογικών πυλών με χρήση της MeMOS λογικής.....	63
3.3.3 Κυκλώματα Αθροιστών.....	65
3.3.4 Πλεονεκτήματα ως προς την λογική συμπερασμού.....	68
3.3.5 Ανάλυση επιδόσεων.....	68
3.3.6 Σύγκριση μεταξύ λογικών οικογενειών.....	73
<b>4. ΣΥΜΠΕΡΑΣΜΑΤΑ.....</b>	<b>76</b>
4.1 Συμπεράσματα.....	76

<b>4.2 Μελλοντικές επεκτάσεις.....</b>	<b>76</b>
<b>ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ.....</b>	<b>78</b>
<b>ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ.....</b>	<b>79</b>
<b>ΑΝΑΦΟΡΕΣ.....</b>	<b>80</b>

## ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Απεικόνιση των έξι συνδυασμών.....	12
Εικόνα 2: Παράδειγμα μίας καμπής τάσης.....	13
Εικόνα 3: Τύποι memristors.....	13
Εικόνα 4: Memristor από διοξείδιο του Τιτανίου.....	14
Εικόνα 5: Συμβατικό κελί CAM 10-T τύπου NOR.....	16
Εικόνα 6: Subthreshold SRAM κελί.....	17
Εικόνα 7: Μία σειρά από 17 memristors διοξειδίου του τιτανίου.....	20
Εικόνα 8: Μία τρισδιάστατη απεικόνιση του Crossbar RRAM.....	21
Εικόνα 9: Τομείς υπεροχής της RRAM τεχνολογίας.....	22
Εικόνα 10: Η κάθετη μνήμη NAND της Samsung.....	23
Εικόνα 11: Η IMPLY λογική πύλη.....	25
Εικόνα 12: Συμπεριφορά μίας ιδανικής IMPLY λογικής πύλης.....	27
Εικόνα 13: IMPLY NAND λογική πύλη.....	29
Εικόνα 14: Μία επέκταση σε μια IMPLY κ-εισόδων NOR.....	30
Εικόνα 15: Βασική δομή ενός memristor-based crossbar.....	32
Εικόνα 16: Sneak paths σε ένα memristive crossbar.....	32
Εικόνα 17: $m \times n$ memristive crossbar.....	33
Εικόνα 18: Μία IMPLY λογική πύλη μέσα σε ένα memristor-based crossbar.....	34
Εικόνα 19: Σχεδιαστική ροή για memristor-based IMPLY λογικές πύλες.....	35
Εικόνα 20: Επιτρεπόμενος χρόνος εγγραφής $T$ .....	38
Εικόνα 21: Επιτρεπόμενες τιμές της $V_{SET}$ .....	38
Εικόνα 22: Tradeoff ανάμεσα στην ταχύτητα και στην ευρωστία.....	39
Εικόνα 23: Η επιτρεπόμενη τιμή της $R_G$ .....	39

Εικόνα 24: Η μεταβλητή κατάσταση της $q$ .....	42
Εικόνα 25: Ο χρόνος εγγραφής μίας IMPLY λογικής πύλης.....	43
Εικόνα 26: Πλήρης αθροιστής.....	47
Εικόνα 27: Ένας πλήρης αθροιστής οχτώ-bit.....	49
Εικόνα 28: Παράδειγμα τοποθέτησης ενός memristor.....	52
Εικόνα 29: MAGIC NOR.....	53
Εικόνα 30: Προσομιώσεις SPICE.....	55
Εικόνα 31: MAGIC NAND πύλη.....	57
Εικόνα 32: MAGIC OR και AND πύλες.....	58
Εικόνα 33: MAGIC NOT πύλη.....	59
Εικόνα 34: Λογικά OR και AND κυκλώματα.....	61
Εικόνα 35: Λογικός υπολογισμός με τη χρήση memristors.....	62
Εικόνα 36: Σχηματικό διάγραμμα μίας NAND πύλης.....	64
Εικόνα 37: Σχηματικό διάγραμμα μιας XOR πύλης.....	64
Εικόνα 38: Η παροδική απόκριση μιας XOR πύλης.....	65
Εικόνα 39: Κύκλωμα ημιαθροιστή.....	66
Εικόνα 40: Εφαρμογή ενός κυκλώματος πλήρους αθροιστή.....	67
Εικόνα 41: Παροδική απόκριση ενός κυκλώματος.....	67
Εικόνα 42: Διάταξη ενός κυκλώματος πλήρους αθροιστή.....	71
Εικόνα 43: Προτεινόμενη αρχιτεκτονική πολλαπλασιαστή.....	72

## ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Πίνακας αλήθειας της IMPLY λειτουργίας.....	26
Πίνακας 2: Τάσεις εισόδου πύλης $V_Q$ και $V_P$ ,.....	37
Πίνακας 3: Χρόνος εγγραφής και ολίσθηση κατάστασης.....	41
Πίνακας 4: Χρόνος εγγραφής για διαφορετικές τιμές της $V_{SET}$ .....	41
Πίνακας 5: Η αντίσταση ενός CMOS driver για 0.12 $\mu\text{M}$ CMOS διαδικασία.....	44
Πίνακας 6: Ολίσθηση τάσης για IMPLY λογική πύλη .....	44
Πίνακας 7: Βασικές Boolean λειτουργίες .....	46
Πίνακας 8: Σύγκριση ανάμεσα στην IMPLY και τη MAGIC.....	54
Πίνακας 9: Παράμετροι memristor (για το μοντέλο VTEAM).....	55
Πίνακας 10: Περίληψη των πυλών MAGIC.....	59
Πίνακας 11: Ανάλυση επίδοσης λογικών πυλών και κυκλωμάτων αθροιστή τα οποία χρησιμοποιούν υβριδική MEMRISTOR-CMOS λογική.....	69
Πίνακας 12: Ανάλυση επίδοσης λογικών πυλών και κυκλωμάτων αθροιστή τα οποία χρησιμοποιούν CMOS λογική.....	69
Πίνακας 13: Αριθμός των MODFET και των memristors για λογικές πύλες με τη χρήση υβριδικής Memristor-CMOS και CMOS λογικής.....	72
Πίνακας 14: Πλήθος MOSFET και Memristor για κύκλωμα αθροιστή το οποίο χρησιμοποιεί υβριδική CMOS-memristor λογική και CMOS λογική.....	73
Πίνακας 15: Σύγκριση παραμέτρων της υβριδικής CMOS-memristor λογικής με τη CMOS λογική.....	73



## ΠΡΟΛΟΓΟΣ

Η παρούσα εργασία είναι καρπός μιας πολύμηνης προσπάθειας στα πλαίσια της πτυχιακής εργασίας του προγράμματος σπουδών του Τμήματος Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών. Μέσω αυτής μου δόθηκε η ευκαιρία να μελετήσω και να εντρυφήσω σε ένα ενδιαφέρον και εφαρμόσιμο πεδίο, αυτό των ολοκληρωμένων κυκλωμάτων. Ευχαριστώ όλους όσους με τον τρόπο τους βοήθησαν ώστε να ολοκληρώσω με επιτυχία το έργο μου, την επιβλέπουσα καθηγήτρια κ. Αγγελική Αραπογιάννη, την οικογένεια μου, αλλά και όλους τους φίλους μου για την τόση δύναμη που μου έδωσαν.

## 1. ΕΙΣΑΓΩΓΗ ΣΤΑ MEMRISTORS

### 1.1 Γενικά

Για περίπου πενήντα χρόνια, τα ολοκληρωμένα ηλεκτρονικά κυκλώματα τα οποία είναι κατασκευασμένα με ημιαγωγικές (semiconductor) συσκευές παρέχουν μια σημαντικά αύξηση στον αριθμό των στοιχείων επεξεργασίας (processing elements) και των bit μνήμης. Αυτή η αύξηση παρέχει επίσης και μεγάλες βελτιώσεις στην ταχύτητα, την κατανάλωση ενέργειας, και την αξιοπιστία, μαζί και με την σημαντική μείωση στο κόστος ανά συσκευή. Αυτές οι τάσεις είναι άμεση συνέπεια της συνεχούς σμίκρυνσης των διαστάσεων των συσκευών στην ημιαγωγική διαδικασία παραγωγής, όπως αρχικά είχε περιγραφεί από τον Gordon Moore το 1965, ο οποίος προέβλεψε την αύξηση και τον πολλαπλασιασμό του digital computing και των εφαρμογών του ("Νόμος του Moore"). Η βασική δομή των digital computing συστημάτων έχει βασιστεί στην κλασική αρχιτεκτονική προγραμματισμού μηχανής η οποία εισήχθη από τον von Neumann το 1940, και χαρακτηρίζεται από τον διαχωρισμό μεταξύ των λειτουργικών μονάδων για εκτέλεση εντολών και αποθήκευση δεδομένων ("αρχιτεκτονική von Neumann").

Εν τούτοις ο νόμος του Moore δεν μπορεί να έχει ισχύ επ' αόριστον. Υπάρχει γενικότερη ομοφωνία στο γεγονός ότι τα νανομεγέθη των CMOS transistor θα φτάσουν τα φυσικά τους όρια μέσα στην επόμενη δεκαετία [1]. Ακόμα και πριν το ενδεχόμενο τέλος του νόμου του Moore λόγω τεχνολογικών ορίων, ο κλάδος της πληροφορικής έχει ήδη αρχίσει να αντιμετωπίζει άλλα θεμελιώδη προβλήματα τα οποία χρήζουν καινοτόμων λύσεων. Ένα από τα πρόβλημα σχετίζεται με την καθυστέρηση και το εύρος το οποίο απαιτείται για τη προσπέλαση την μνήμης, και είναι ευρύτερα γνωστό ως "το τείχος της μνήμης" ("the memory wall") [2].

Στα μελλοντικά χρόνια, όταν τα μεγέθη των συσκευών δεν θα μπορούν να σμικρυνθούν περαιτέρω, η μικροηλεκτρονική τεχνολογία θα χρειαστεί καινοτομίες "υπεράνω του Moore" για να υποστηρίξει νέες εφαρμογές. Αυτές οι νέες τεχνολογίες μπορεί να περιλαμβάνουν νέες επαναστατικές συσκευές όπως carbon nanotube ή spintronic devices. Μία λιγότερο ριζική υβριδική προσέγγιση, η οποία θα περιλαμβάνει την CMOS τεχνολογία σε συνδυασμό με νέες, θεωρείται ότι θα αποτελέσει μια πρακτική λύση για τα επόμενα είκοσι με τριάντα χρόνια. Μία από αυτές τις τεχνολογίες η οποία επεκτείνει τις δυνατότητες των CMOS είναι και οι memristive συσκευές. Αυτή η εργασία επικεντρώνεται στις memristive τεχνολογίες στον αντίκτυπο που θα χουν στους μελλοντικούς υπολογιστές.

### 1.2 Memristors

Τα τελευταία είκοσι πέντε χρόνια, η μνήμη τύπου flash η οποία βασίζεται στην παγίδευση τάσης στα MOS τρανζίστορ έχει σμικρυνθεί δραματικά, ξεπερνώντας ακόμα και τον νόμο του Moore. Μεγέθη όμως κάτω των 20 nm επιφέρουν και βασικές προκλήσεις, ειδικότερα στον ρυθμό λάθος bit (error bit rate) και μια σημαντική μείωση στην αντοχή γραψίματος (ο αριθμός των κύκλων γραψίματος πριν η μνήμη γίνει αναξιόπιστη). Αυτές οι προκλήσεις γίνονται μη αποδεκτές, όταν η τεχνολογία flash διαδικασίας χρησιμοποιείται σε μεγέθη κάτω των 15 nm [3].

Τα τελευταία χρόνια, πολλές εναλλακτικές τεχνολογίες έχουν ανακαλυφθεί προς αντικατάσταση της flash. Για τις περισσότερες από αυτές τις υποψήφιας τεχνολογίες, τα αποθηκευμένα δεδομένα παριστάνονται ως μια αντίσταση και η συσκευή αποθήκευσης κατασκευάζεται ανάμεσα στα μεταλλικά στρώματα. Όλες αυτές οι τεχνολογίες μοιράζονται κοινές ιδιότητες όπως μη μεταβλητότητα, σχετικά υψηλή αντοχή γραψίματος, υψηλή πυκνότητα, καλή επεκτασιμότητα κάτω από τα 10 nm, και γρήγορο διάβασμα και γράψιμο. Ορισμένες αναδυόμενες τεχνολογίες μνήμης έχουν σαφή ταχύτητα και αντοχή για να θεωρηθούν αντικαταστάτριες των SRAM και DRAM διευκολύνοντας έτσι την χρήση μίας και μόνο γενικής μνήμης [4]. Αυτές οι αναδυόμενες τεχνολογίες μνήμης μόνιμης αποθήκευσης μπορούν να χαρακτηριστούν ως memristors, η ακριβέστερα memristive συσκευές.

### 1.3 Η θεωρία των memristors

Το 1971, ο Leon Chua παρατήρησε την ανάγκη για ένα νέο θεμελιώδες στοιχείο κυκλώματος (circuit element) εκτός των αντιστάτη, πυκνωτή και πηνίου [5]. Ο Chua αιτιολόγησε την ύπαρξη ενός χαμένου στοιχείου και λόγω συμμετρίας αν κοιτάξουμε τους 6 πιθανούς συνδυασμούς στις σχέσεις των τεσσάρων βασικών μεταβλητών κυκλώματος, της τάσης  $V$ , της έντασης  $I$ , της ροής  $\varphi$ , και του ηλεκτρικού φορτίου  $q$ . Οι αντιστάτες συνδέουν την τάση με την ένταση μέσω του νόμου του Ohm ( $V = IR$ ), οι πυκνωτές συνδέουν το ηλεκτρικό φορτίο με την τάση ( $q = CV$ ), και τα πηνία συνδέουν την ένταση με τη ροή ( $\varphi = LI$ ). Η έκτη πιθανή σχέση είναι η σύνδεση ανάμεσα στο ηλεκτρικό φορτίο και την ροή και δεν καλύπτεται από κανένα από τα συμβατικά στοιχεία των κυκλωμάτων. Ο Chua επίσης ισχυρίστηκε και για το καλό της πληρότητας την ύπαρξη και ενός τέταρτου θεμελιώδους στοιχείου κυκλώματος το οποίο θα συνδέει το φορτίο με τη ροή και το ονόμασε memristor σε συντομία του 'memory resistor'. Ο τύπος για ένα memristor ελεγχόμενο από φορτίο είναι

$$v(t) = M(q(t))i(t),$$

όπου

$$M(q(t)) = d\varphi(q)/dq.$$

Παρόμοια, για ένα memristor ελεγχόμενο από τη ροή είναι

$$i(t) = W(\varphi(t))v(t),$$

όπου

$$W(\varphi(t)) = dq(\varphi)/d\varphi.$$

Η μεταβλητή  $M(q(t))$  γνωστή και ως memristance έχει τις μονάδες αντίστασης και βασίζεται στην ενσωμάτωση της έντασης που διαπερνά μία συσκευή. Για τον λόγο αυτό το memristor είναι ουσιαστικά ένα παθητικό δίθυρο στοιχείο με μεταβλητή αντίσταση, η οποία αλλάζει ανάλογα το ιστορικό της συσκευής.

Το 1976, η θεωρία των memristors επεκτάθηκε από τον Chua και τον Kang σε ένα μη

γραμμικό δυναμικό σύστημα το οποίο ονομάστηκε memristive σύστημα [6]. Παρόμοια με τα memristor, μια memristive συσκευή είναι μία παθητική δύο τερματικών συσκευή με μεταβλητή αντίσταση.

Ο τύπος μίας ελεγχόμενης από την ένταση χρονικά αναλλοίωτης memristive συσκευής είναι

$$v(t) = M(x,i)i(t),$$

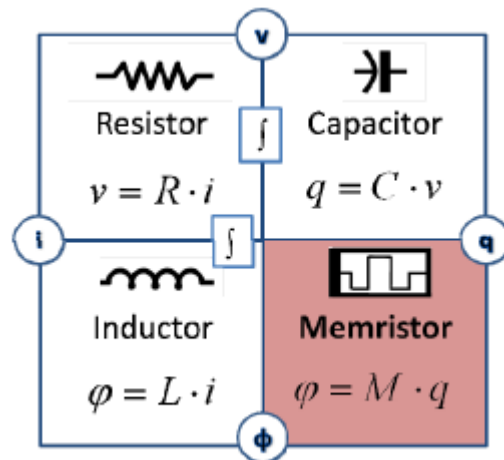
$$dx/dt = f(x,i),$$

όπου  $M(x,i)$  η memristance της συσκευής. Παρόμοια, ο τύπος μίας ελεγχόμενης από την τάση χρονικά αναλλοίωτης συσκευής είναι

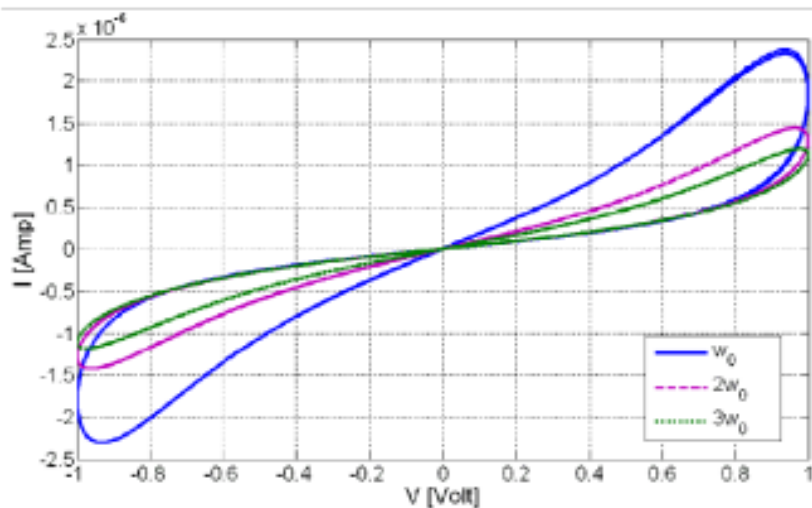
$$i(t) = W(x,i)v(t),$$

$$dx/dt = f(x,v).$$

Τα memristor και οι memristive συσκευές εμφανίζουν υστέρηση στην καμπή τάσης-έντασης. Η υστέρηση αυτή βασίζεται στην εισαγωγή, όπου για εισαγωγή υψηλών συχνοτήτων η συσκευή συμπεριφέρεται ως ένας γραμμικός αντιστάτης.

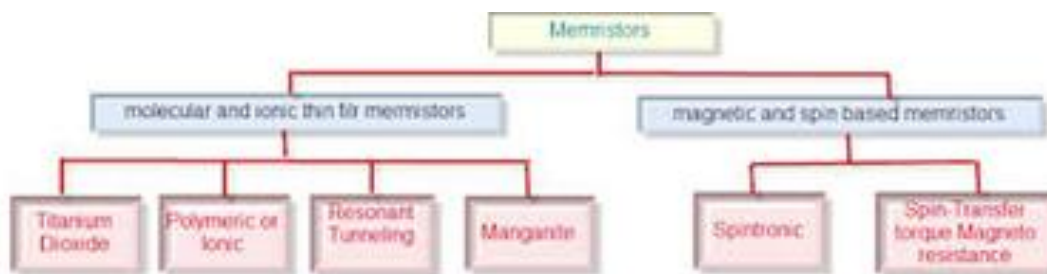


**Εικόνα 1:** Απεικόνιση των έξι συνδυασμών των σχέσεων ανάμεσα στην τάση  $v$ , την ένταση  $i$ , την ροή  $\varphi$ , και το ηλεκτρικό φορτίο  $q$ . Το memristor συνδέει τη ροή με το φορτίο.



Εικόνα 2: Παράδειγμα μίας καμπής τάσης - έντασης των memristors για διαφορετικές συχνότητες εισαγωγής.

Τα memristors μπορούν να είναι πολλών διαφορετικών τύπων ανάλογα με τον τρόπο κατασκευής τους.



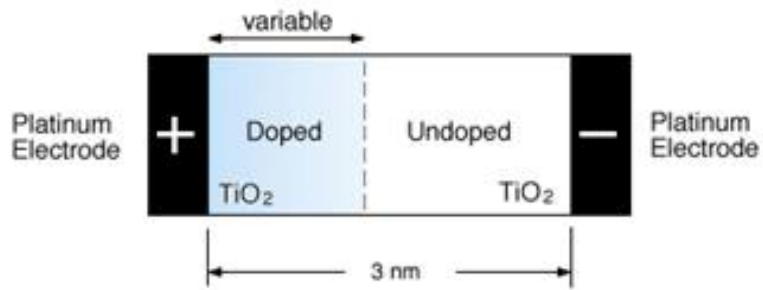
Εικόνα 3: Τύποι memristors

Τα memristors από λεπτή στρώση διοξειδίου του Τιτανίου (Titanium dioxide) ήταν τα πρώτα τα οποία κατασκευάστηκαν και εξελίχθηκαν για μοντελισμό και σχεδίαση. Στα πολυμερικά ή ιονικά memristors τα στερεάς κατάστασης ιόντα (είτε κατιόντα είτε ανιόντα) μετακινούνται κατά μήκος της κατασκευής ως μεταφορείς φορτίου resonant-tunneling diode memristors και χρησιμοποιούν ενυσιχιμένες κβαντικές διόδους. Τα μαγνητιώδη memristors χρησιμοποιούν ένα

υπόστρωμα από οξειδίο δύο στρώσεων το οποίο βασίζεται στο μαγγάνιο σε αντίθεση με το διοξειδίο του τιτανίου. Στα spintronic memristors, η κατεύθυνση της περιστροφής των ηλεκτρονίων αλλάζει τη μαγνητική κατάσταση της συσκευής οποία αλλάζει και την αντίσταση της. Στα spin-transfer torque memristors, η σχετική μαγνητική ευθυγράμμιση των δύο

Εφαρμογές memristors σε VLSI κυκλώματα

ηλεκτροδίων επηρεάζει τη μαγνητική κατάσταση της μαγνητικής σύνδεσης η οποία με τη σειρά της αλλάζει και την αντίσταση της.



Εικόνα 4: Memristor από διοξείδιο του Τιτανίου

## 2. ΕΦΑΡΜΟΓΕΣ MEMRISTORS ΣΕ ΜΝΗΜΕΣ

### 2.1 Μνήμη MCAM

Η αλματώδης πρόοδος των δικτύων όπως του διαδικτύου έχει επιφέρει την ανάγκη για δημιουργία νέων εξαρτημάτων και σχετικών κυκλωμάτων τα οποία θα είναι συμβατά με την CMOS τεχνολογία. Όσο ο νόμος του Moore γίνεται όλο και πιο δύσκολος στην υλοποίηση του, η ενσωμάτωση εντελώς νέων και διαφορετικών τεχνολογιών όπως τα spintronics, τα carbon nano tube field transistors (CNFET), τα οπτικά νανοκυκλώματα, και πιο πρόσφατα τα memristors, κεντρίζουν στο ενδιαφέρον για την δημιουργία νέων κυκλωμάτων και συστημάτων στο System on System (SoS) domain.

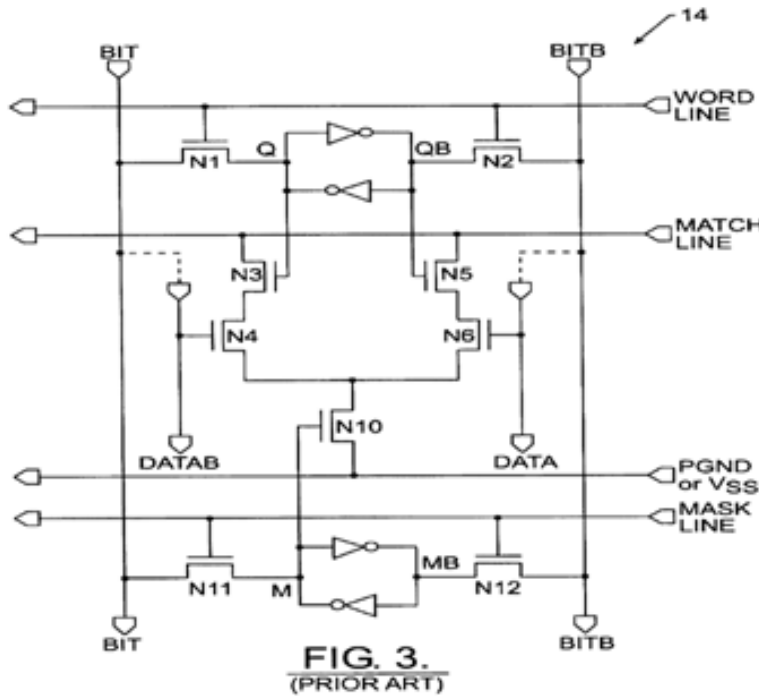
Εδώ θα ασχοληθούμε με τον σχεδιασμό και τη μοντελοποίηση του κελιού μνήμης ως μέρος μίας μνήμης διευθυνσιοδότησης περιεχομένου (MCAM) η οποία θα αποτελείται από έναν συνδυασμό memristor και n-τύπου MOS συσκευές. Ένα τυπικό κελί CAM σχηματίζει ένα κελί SRAM το οποίο αποτελείται από 2 n-τύπου και 2 p-τύπου MOS transistors. Η κατασκευή ενός SRAM κελιού το οποίο θα εκμεταλλεύεται τη τεχνολογία των memristors και θα έχει συμπεριφορά μνήμης μόνιμης αποθήκευσης (NVM) καθώς και θα μπορεί να κατασκευαστεί ως μια επέκταση της CMOS τεχνολογίας αποτελεί την βασική ιδέα μιας έρευνας προς την κατεύθυνση της μείωσης της κατανάλωσης της ηλεκτρικής ενέργειας.

Η σχεδίαση ενός κελιού CAM βασίζεται στο τέταρτο παθητικό στοιχείο κυκλώματος το memristor το οποίο ανακαλύφθηκε από τον Chua. Το memristor συμπεριφέρεται σαν ένας διακόπτης, ακριβώς όπως ένα transistor. Παρόλα αυτά όμως σε αντίθεση με το transistor, έχει δύο τερματικά αντί για τρία και δεν χρειάζεται ενέργεια για να διατηρήσει οποιαδήποτε από τις δύο καταστάσεις του. Σημειώστε ότι το memristor αλλάζει την αντίσταση του μεταξύ δύο τιμών και αυτό επιτυγχάνεται μέσω της κίνησης ενός ιονικού φορτίου μέσα σε μια στρώση οξειδίου, καθώς και ότι αυτές οι καταστάσεις αντίστασης είναι μόνιμες. Αυτό είναι ένα ιδιαίτερα σημαντικό χαρακτηριστικό το οποίο επηρεάζει σε μεγάλο βαθμό την αρχιτεκτονική των συστημάτων CAM, καθώς μπορεί να διακοπή η παροχή ρεύματος στα CAM μπλοκ χωρίς να χαθούν τα αποθηκευμένα δεδομένα. Για το λόγο αυτό τα κελιά CAM τα οποία είναι βασισμένα σε memristors έχουν την δυνατότητα εξοικονόμησης ενός σημαντικού ποσοστού ενέργειας. Για να γίνει καλύτερα κατανοητή η ανάγκη χρήσης memristors αλλά και τα πλεονεκτήματα που αυτά προσφέρουν θα κάνουμε τώρα μία σύντομη περίληψη ενός συμβατικού CAM κελιού το οποίο χρησιμοποιεί SRAM όπως αυτό παρουσιάζεται στην εικόνα 4. Οι δύο αντιστροφείς χρησιμοποιούν 4 transistors τα οποία περιλαμβάνουν 2 τύπου p transistor τα οποία χρειάζονται περισσότερη περιοχή πυριτίου. Προβλήματα όπως σχετικά υψηλό ρεύμα διαρροής για CMOS νανοτεχνολογίας και η ανάγκη για συμπερίληψη τόσο των  $V_{DD}$  γραμμών όσο και των γραμμών γείωσης σε κάθε κελί φέρνουν τους σχεδιαστές των CAM αντιμέτωπους με νέες προκλήσεις, έτσι ώστε να αυξηθεί η πυκνότητα και να διατηρηθεί μία λογική κατανάλωση ενέργειας. Έτσι για να ικανοποιηθεί ο συνδυασμός υπερ-πυκνών σχεδίων, χαμηλής κατανάλωσης και υψηλών επιδόσεων το κελί SRAM είναι στο επίκεντρο των αρχιτεκτονικών σχεδίων.

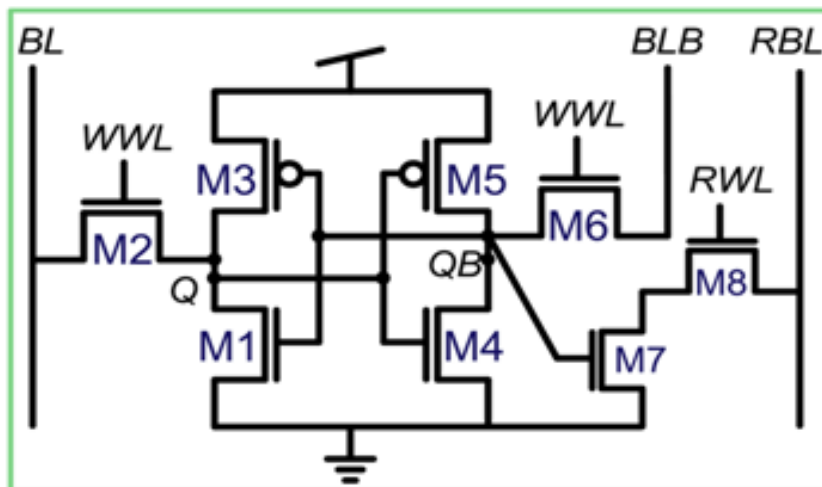
Για παράδειγμα, ένα από τα πιο γνωστά προβλήματα του συμβατικού 6-T SRAM για εφαρμογές πολύ χαμηλής ενεργειακής κατανάλωσης είναι το περιθώριο στατικού θορύβου (static noise margin) (SNM) του. Η βασική τεχνική που χρησιμοποιείται για τη σχεδίαση μίας

μνήμης υπέρ-χαμηλής

κατανάλωσης είναι η κλιμάκωση της τάσης η οποία φέρνει την λειτουργία του CMOS στη περιοχή κατωφλίου. Ο Verma και ο Chandrakasan υποστήριξαν ότι στη παροχή πολύ χαμηλών τάσεων το περιθώριο στατικού θορύβου για SRAM θα εξαφανιστεί λόγω της παραλλαγής της διαδικασίας. Για την θεωρία τους αυτή ο Verma και Chandrakasan πρότειναν ένα 8-T SRAM κελί το οποίο φαίνεται στην εικόνα 5. Αυτό σημαίνει ότι υπάρχει ανάγκη για σημαντική αύξηση της περιοχής πυριτίου όταν η τάση αποκλιμακώνεται.



Εικόνα 5: Συμβατικό κελί CAM 10-T τύπου NOR



Εικόνα 6: Subthreshold SRAM κελί



Η αποτυχία είναι ένα πολύ μεγάλο ζήτημα στο σχεδιασμό των πολύ πυκνών μνημών (μνήμες με μεγάλη χωρητικότητα). Για το λόγο αυτό συνήθως χρησιμοποιείται ένα μεγάλο εύρος τεχνικών ανοχής βλαβών [7]. Μία παραδοσιακή αντιμετώπιση βλαβών οι οποίες προέρχονται από την SRAM είναι και η αντιγραφή κελιών μνήμης. Αυτή η λύση προφανώς προκαλεί μια σημαντική αύξηση της περιοχής πυριτίου, η οποία με τη σειρά της επιδεινώνει και το θέμα της κατανάλωσης ενέργειας.

Μερικά από τα συγκεκριμένα CAM κελιά, για παράδειγμα, η μνήμη διευθυνσιοδότησης τριαδικού περιεχομένου (TCAM) συνήθως χρησιμοποιείται για το σχεδιασμό πολύ γρήγορων εφαρμογών αναζήτησης σε ρούτερ δικτύων, όπως η προώθηση πακέτων.

Συμπερασματικά, η ιδέα ενός στοιχείου κυκλώματος το οποίο σχετίζει το φορτίο με την μαγνητική ροή υλοποιήσιμου σε νανοκλίμακα το οποίο θα έχει τη δυνατότητα να θυμάται τη προηγούμενη ιστορία του, δημιουργεί ενδιαφέρουσες προσεγγίσεις στις μελλοντικές αρχιτεκτονικές που θα βασίζονται σε CAM. Η κλιμάκωση της CMOS τεχνολογίας γίνεται ιδιαίτερα προκλητική όταν πέφτει κάτω από τα 10 nm για αυτό το λόγο τα νανο-χαρακτηριστικά των memristors μπορούν να αξιοποιηθούν σε σημαντικό βαθμό. Έτσι το memristor μπορεί να γίνει ένας ιδιαίτερα δυνατός υποψήφιος για τις μελλοντικές tera-bit μνήμες.

Τα χαρακτηριστικά μόνιμης αποθήκευσης και η γεωμετρία νανοκλίμακας σε συνδυασμό με τη συμβατότητα με τη CMOS τεχνολογία αυξάνει τη πυκνότητα σε κελιά μνήμης, μειώνει την κατανάλωση ενέργειας και προσδίδει νέες προσεγγίσεις στη διαχείριση της μνήμης μέσα από την απενεργοποίηση των μπλοκ των MCAM κελιών χωρίς την απώλεια αποθηκευμένων δεδομένων.

## 2.2 Resistive RAM

Η ωμική μνήμη τυχαίας προσπέλασης (resistive random access memory) (RRAM η ReRAM) είναι ένας τύπος μνήμης τυχαίας προσπέλασης μόνιμης αποθήκευσης η οποία λειτουργεί με το να αλλάζει την αντίσταση σε ένα διηλεκτρικό στερεάς κατάστασης υλικό το οποίο συχνά αναφέρεται ως memristor. Αυτή η τεχνολογία έχει μερικές ομοιότητες με τη CBRAM και τη μνήμη αλλαγής φάσης (phase change memory) (PCM).

Η CBRAM περιλαμβάνει ένα ηλεκτρόδιο το οποίο παρέχει ιόντα τα οποία διαλύονται εύκολα σε ένα ηλεκτρολυτικό υλικό. Η PCM παράγει μεγάλες θερμικές απώλειες για να επηρεάσει άμορφες σε κρυσταλλικές η κρυσταλλικές σε άμορφες αλλαγές φάσης. Σε αντίθεση, η RRAM δημιουργεί ανωμαλίες σε ένα λεπτό στρώμα οξειδίου, γνωστές ως κενές θέσεις οξυγόνου οι οποίες ακολούθως μπορούν να φορτιστούν και να ολισθήσουν κάτω από ένα ηλεκτρικό πεδίο. Η κίνηση των ιόντων και των κενών θέσεων οξυγόνου στο οξείδιο είναι ανάλογες της κίνησης των ηλεκτρονίων και των τρυπών σε έναν ημιαγωγό.

Η RRAM είναι υπό ανάπτυξη από πολλές εταιρίες, μερικές από τις οποίες έχουν και

συμπληρώσει αιτήσεις για πατέντες πάνω σε διάφορες εφαρμογές αυτής της τεχνολογίας. Αν και συχνά αντιμετωπίζεται ως αντικαταστάτρια τεχνολογία για τη μνήμη τύπου flash, τα πλεονεκτήματα σε κόστος και απόδοση της RRAM δεν έχουν γίνει ακόμα αρκετά ορατά στις περισσότερες εταιρίες έτσι ώστε να προχωρήσουν στην αντικατάστασή τους. Μια μεγάλη γκάμα υλικών μπορούν να χρησιμοποιηθούν στην κατασκευή των RRAM. Ωστόσο η πρόσφατη ανακάλυψη ότι το διάσιμο διηλεκτρικό  $\text{HfO}_2$  μπορεί να χρησιμοποιηθεί ως μια RRAM χαμηλής τάσης έχει ενθαρρύνει πολλές εταιρίες να κάνουν έρευνες προς αυτήν την κατεύθυνση. Ακόμα πιο πρόσφατα το  $\text{SiO}_x$  έχει παρατηρηθεί να προσφέρει σημαντικά πλεονεκτήματα. Η Weebit-Nano Ltd είναι μία εταιρία η οποία εργάζεται πάνω στη χρήση  $\text{SiO}_x$  και έχει ήδη παρουσιάσει λειτουργικές συσκευές.

Τον Φεβρουάριο του 2012 η Rambus μία αμερικάνικη εταιρία τεχνολογίας εξαγόρασε την Unity Semiconductor μια εταιρία με εξειδίκευση στην ωμική μνήμη τυχαίας προσπέλασης προς 35 εκατομμύρια δολάρια. Τον Μάιο του 2012 η Panasonic έβγαλε στην αγορά ένα kit αξιολόγησης RRAM το οποίο βασιζόταν σε μια αρχιτεκτονική κελιού μνήμης ενός τρανζίστορ-ενός αντιστάτη. Το 2013, η Crossbar παρουσίασε ένα πρωτότυπο ωμικής μνήμης τυχαίας προσπέλασης ως ένα τσιπ το οποίο είχε το μέγεθος ενός γραμματοσήμου και είχε τη δυνατότητα αποθήκευσης ενός TeraByte δεδομένων. Τον Αύγουστο του 2013, η εταιρία δήλωσε το ξεκίνημα μίας μεγάλης κλίμακας παραγωγής του RRAM chip της μέχρι το 2015. Η δομή της μνήμης μοιάζει πολύ αυτή με αυτή μια CBRAM με βάση τον άργυρο.

Κατά καιρούς πολλές διαφορετικές μορφές της RRAM έχουν αποκαλυφθεί στο κοινό, βασιζόμενες και σε ποικιλία διαφορετικά διηλεκτρικά υλικά. Το διοξείδιο του πυριτίου μάλιστα είχε ήδη γίνει γνωστό ότι παρουσιάζει αντίσταση μεταγωγής από το μακρινό 1967, και πρόσφατα ξαναήρθε και πάλι στο προσκήνιο.

Ο Leon Chua υποστήριξε ότι όλες οι συσκευές μνήμης μόνιμης αποθήκευσης δύο τερματικών συμπεριλαμβανομένης της ωμικής μνήμης τυχαίας προσπέλασης θα πρέπει να θεωρούνται memristors. Ο Stan Williams από τα εργαστήρια της HP επίσης δήλωσε ότι η RRAM είναι ένα memristor. Παρόλα αυτά υπάρχουν πολλοί που αμφισβητούν αυτήν την ορολογία αλλά και την εφαρμογή της θεωρίας των memristor σε οποιαδήποτε συσκευή.

Το 2014 ερευνητές ανακοίνωσαν την κατασκευή μιας συσκευής η οποία χρησιμοποιούσε ένα πορώδες διηλεκτρικό οξειδίο του πυριτίου. Η κατασκευή αυτή ήταν αποτέλεσμα της ανακάλυψης μονοπατιών αγωγισμού νήματος το 2010.

Η Βασική ιδέα της δομής της ωμικής μνήμης τυχαίας προσπέλασης είναι ότι ένα διηλεκτρικό το οποίο υπό κανονικές συνθήκες είναι μονωτικό μπορεί να γίνει αγωγίμο μέσα από ένα μονοπάτι νήματος ή αγωγιμότητας το οποίο δημιουργείται ύστερα από την εφαρμογή μιας σημαντικά μεγάλης τάσης. Το μονοπάτι αγωγιμότητας μπορεί να δημιουργηθεί από διάφορους μηχανισμούς. Μόλις το νήμα δημιουργηθεί, είναι σε θέση να επαναοριστεί (να σπάσει, καταλήγοντας σε υψηλότερη αντίσταση) ή να οριστεί (καταλήγοντας σε μικρότερη αντίσταση) από διαφορετική τάση. Πολλές διαδρομές ρεύματος, αντί μόνο ενός μόνο νήματος μπορούν να επιτευχθούν.

Όσο αναφορά τις μορφές λειτουργίας, για τις μνήμες τυχαίας προσπέλασης, προτιμάται η αρχιτεκτονική 1T1R (ένα transistor, ένας αντιστάτης) λόγω του ότι το transistor απομονώνει φορτίο μόνο στα κελιά τα οποία έχουν επιλεχθεί. Στον αντίποδα, η αρχιτεκτονική cross-point είναι πιο συμπαγής και καθιστά ικανή την δημιουργία στρωμάτων μνήμης κάθετης στοίβας, τα

οποία είναι ιδανικά για συσκευές μαζικής αποθήκευσης. Ωστόσο, χωρίς την παρουσία transistor, η απομόνωση μπορεί να επιτευχθεί από μία συσκευή επιλογέα, όπως μια δίοδος σε σειρά με το στοιχείο μνήμης. Οι δυνατότητες απομόνωσης αυτού του είδους είναι κατώτερες από αυτές με τη χρήση transistor εάν η λειτουργία on/off του επιλογέα δεν είναι επαρκής, οριοθετώντας την ικανότητα της μνήμης να διαχειριστεί πολύ μεγάλες συστοιχίες (arrays) σε αυτή την αρχιτεκτονική. Ένας διακόπτης κατωφλίου βασισμένος σε λεπτή ταινία μπορεί να λειτουργήσει σαν επιλογέας για διπολικές και μονοπολικές RRAM.

Πολλές δημοσιεύσεις εργασιών έχουν γίνει γύρω από την ωμική μνήμη τυχαίας προσπέλασης. Μία από αυτές ήταν στο συνέδριο IEDM το 2007, όπου για πρώτη φορά προτάθηκε ότι η RRAM παρουσιάζει χαμηλότερο ρεύμα προγραμματισμού από τις PRAM και MRAM χωρίς να θυσιάζει στο ελάχιστο την προγραμματιστική απόδοση ή αντοχή. Στις 30 Απριλίου του 2008, η HP ανακοίνωσε ότι είχε ανακαλύψει το memristor, το οποίο είχε οραματιστεί αρχικά ο Chua το 1971 ως το τέταρτο χαμένο βασικό στοιχείο κυκλώματος. Στις 8 Ιουλίου ανακοίνωσαν ότι θα αρχίσουν τη δημιουργία RRAM πρωτοτύπων με τη χρήση memristors. Στο IEDM του 2008, η υψηλότερης απόδοσης RRAM τεχνολογία μέχρι εκείνη τη στιγμή παρουσιάστηκε από την ITRI, επιδεικνύοντας χρόνους εναλλαγής μικρότερους από 10 ns και ρεύματα μικρότερα από 30 mA. Στο IEDM του 2010 η ITRI έσπασε ξανά το ρεκόρ ταχύτητας με χρόνο εναλλαγής μικρότερο από 0.3 ns αλλά και βελτιώσεις στον τομέα της κατεργασίας και λειτουργίας. Το 2012 η IMEC παρουσίασε νέες ενημερώσεις στο RRAM πρόγραμμα της στο συμπόσιο για την VLSI τεχνολογία και κυκλώματα, μεταξύ των οποίων και μία λύση με ρεύμα λειτουργίας 500 nA.



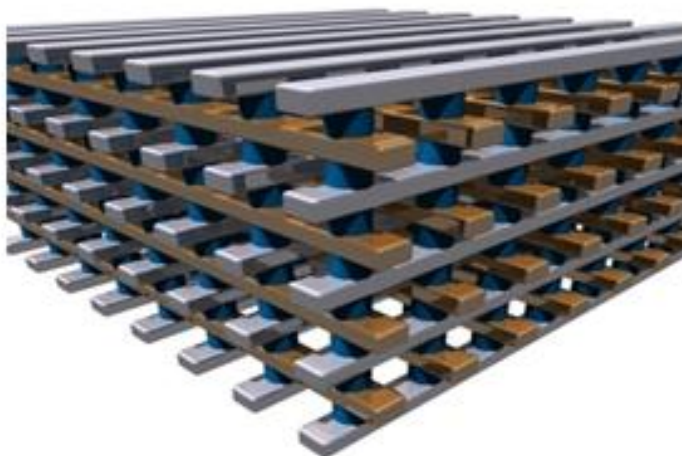
Εικόνα 7: Μία σειρά από 17 memristors διοξειδίου του τιτανίου από τα εργαστήρια της HP

### 2.3 Σύγκριση RRAM με 3D NAND flash

Σε μερικά χρόνια είναι πολύ πιθανή η κατασκευή smartphone, tablet ή laptop με εκατοντάδες gigabytes ή ακόμα και terabytes από γρήγορη μόνιμης αποθήκευσης μνήμη, χάρις σε δύο νέες τεχνολογίες.

Πρώτη η Samsung ανακοίνωσε την ευρεία παραγωγή τρισδιάστατων κατακόρυφων NAND chips (V-NAND chips) και αργότερα η start-up Crossbar ανακοίνωσε και αυτή με τη σειρά της το πρωτότυπο της για chip ωμικής μνήμης τυχαίας προσπέλασης (RRAM).

Η τρισδιάστατη NAND τεχνολογία παίρνει τα σημερινά flash chips, τα οποία κατασκευάζονται σε οριζόντιο επίπεδο και τα γυρίζει πλάγιως. Ύστερα σαν μικροσκοπικοί ουρανοξύστες μνήμης, τα στοιβάζει το ένα δίπλα στο άλλο για να δημιουργήσει ένα πολύ πιο πυκνό chip με τη διπλάσια απόδοση γραψίματος και δέκα φορές πιο αξιόπιστο από τα σημερινά δισδιάστατα NAND chips.



**Εικόνα 8: Μία τρισδιάστατη απεικόνιση του Crossbar RRAM τεχνολογίας**

Η flash NAND χρησιμοποιεί τρανζίστορ για να αποθηκεύσει ένα bit πληροφορίας σε ένα κελί πυριτίου, ενώ η RRAM χρησιμοποιεί μικροσκοπικά αγωγίμα νήματα τα οποία συνδέουν στρώματα πυριτίου για να αναπαραστήσουν ένα bit πληροφορίας ένα ψηφιακό άσσο ή μηδενικό. Στην RRAM, το στρώμα νιτριδίου του πυριτίου το οποίο βρίσκεται στην κορυφή δημιουργεί ένα αγωγίμο ηλεκτρόδιο, ενώ το κατώτατο στρώμα είναι μη αγωγίμο οξειδίο του πυριτίου. Ένα θετικό φορτίο δημιουργεί μια σύνδεση μέσω νήματος μεταξύ των δύο στρωμάτων πυριτίου, το οποίο αναπαριστά έναν ψηφιακό άσσο. Ένα αρνητικό φορτίο σπάει αυτή τη σύνδεση δημιουργώντας μία στρώση αντίστασης ή ένα μηδενικό.

Στο ερώτημα ποια από τις δύο τεχνολογίες θα κυριαρχήσει στο άμεσο μέλλον δεν υπάρχει άμεση απάντηση καθώς οι ειδικοί δεν μπορούν να πουν με σιγουριά κατά πόσο η τρισδιάστατη NAND flash μπορεί να επιμηκύνει τη ζωή της τρέχουσας NAND flash τεχνολογίας. Μερικοί υποστηρίζουν ότι από τα 24 στρώματα τα οποία χρησιμοποιεί σήμερα η Samsung θα ξεπεράσει τα 100 στο μέλλον, άλλοι πιστεύουν η τεχνολογία έχει ακόμα μόνο δύο ή τρεις γενιές εξέλιξης, θεωρώντας ότι θα αντιμετωπίσει τεράστιες δυσκολίες όταν φτάσεις τα 64 στρώματα.

Σε αντίθεση, η RRAM ξεκινάει από την αρχή με ένα σοβαρό πλεονέκτημα. Είναι πιο πυκνή από την NAND με καλύτερες επιδόσεις και μεγαλύτερη αντοχή. Αυτό σημαίνει ότι η RRAM θα είναι σε θέση να χρησιμοποιεί στρώσεις πυριτίου οι οποίες θα έχουν το μισό μέγεθος εκείνων που χρησιμοποιούν σήμερα οι κατασκευαστές των σημερινών NAND flash μνημών. Μία σημαντική επίσης λεπτομέρεια αποτελεί το γεγονός ότι τα σημερινά εργοστάσια παραγωγής flash μνημών δεν θα χρειαστεί να κάνουν ριζικές αλλαγές στον εξοπλισμό τους σύμφωνα με τον διευθυντή της Crossbar George Minassian. Ο ίδιος δήλωσε ότι το κόστος εισαγωγής της νέας τεχνολογίας στις μονάδες παραγωγής δεν θα ξεπεράσει τα δύο με τρία εκατομμύρια δολάρια δηλαδή όσο το κόστος ανάπτυξης μιας νέας γραμμής παραγωγής για μια νέα NAND flash μνήμη.

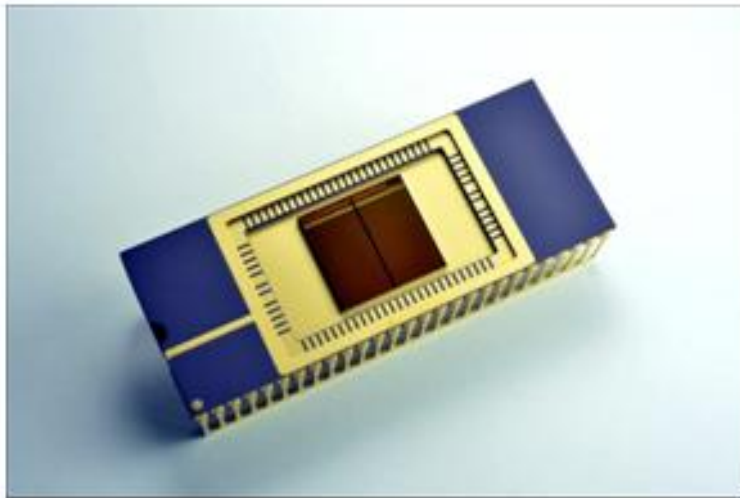


Εικόνα 9: Τομείς υπεροχής της RRAM τεχνολογίας

Η Crossbar υποστηρίζει ότι η RRAM τεχνολογία της έχει χρόνο επεξεργασίας μόλις 30 ns σε αντίθεση με την κορυφαία flash μνήμη της Samsung της οποίας ο χρόνος είναι περίπου στα 0.057 ms. Σύμφωνα με τον Minassian η ωμική μνήμη μπορεί να αντέξει περίπου 10.000 κύκλους εγγραφής-διαγραφής αριθμός ο οποίος είναι λίγο μεγαλύτερος από αυτόν μιας τυπικής NAND flash μνήμης της σύγχρονης αγοράς. Ακόμα η Crossbar προσδοκεί την μαζική παραγωγή του RRAM τσιπ της μέσα στα επόμενα δύο χρόνια. Ο Minassian υποστήριξε ότι η εταιρία του έχει ήδη έρθει σε συμφωνία με ένα εργοστάσιο παραγωγής flash μνημών για την κατασκευή των τσιπ.

Η RRAM καθώς και η τρισδιάστατη NAND προμηνύουν τρομερά τεχνολογικά άλματα τόσο στις επιδόσεις της μνήμης όσο και στο μέγεθος αποθήκευσης. Η RRAM της Crossbar υπόσχεται 20 φορές ταχύτερους χρόνους εγγραφής (writing times) και 10 φορές περισσότερη

αντοχή από την σημερινή επίπεδη NAND. Όπως τα τρισδιάστατα NAND έτσι και τα RRAM τσιπ μνήμης θα κατασκευαστούν σε στοίβες κάνοντας έτσι δυνατή τη δημιουργία μίας μνήμης χωρητικότητας 1 terabyte με περίπου το μισό μέγεθος μιας NAND flash μνήμης με παρόμοια χωρητικότητα. Μάλιστα το αρχικό RRAM τσιπ της Crossbar θα είναι ικανό να αποθηκεύει έως και 1 terabyte δεδομένων έχοντας μέγεθος μικρότερο από ενός γραμματισήμου. Επιπλέον όσο αναφορά τις επιδόσεις η RRAM έχει ένα ακόμα πλεονέκτημα. Ένα NAND flash τσιπ έχει σήμερα ταχύτητες εγγραφής της τάξης των 7 MB/sec. Οι SSD και οι κάρτες flash μπορούν να επιτύχουν ταχύτητες 400MB/sec αλλά τρέχοντας πολλά παράλληλα τσιπ. Ένα RRAM τσιπ μπορεί να επιτύχει μέχρι και 140 MB/sec αλλά χωρίς το παράλληλο τρέξιμο πολλών τσιπ.



**Εικόνα 10: Η κάθετη μνήμη NAND της Samsung**

### 3. ΧΡΗΣΗ MEMRISTORS ΣΕ ΛΟΓΙΚΕΣ ΠΥΛΕΣ

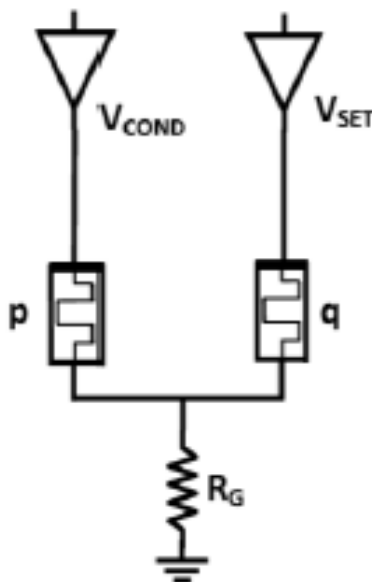
#### 3.1 Λογική υλικού συμπερασμού βασισμένη σε memristors (IMPLY)

Η χρήση memristors για την εκτέλεση λογικών λειτουργιών έχει προταθεί με πολλούς και διαφορετικούς τρόπους. Σε μερικές λογικές οικογένειες, τα memristors συνεργάζονται με CMOS κατασκευές για να εκτελέσουν μια λογική λειτουργία, ενώ οι λογικές τιμές αναπαριστώνται ως επίπεδα τάσης. Στην υβριδική memristor-CMOS λογική οικογένεια τα memristor λειτουργούν ως υπολογιστικά στοιχεία εκτελώντας OR και AND Boolean λειτουργίες, ενώ τα CMOS transistors εκτελούν λογική αναστροφή και ενίσχυση των λογικών σημάτων τάσης.

Μία ακόμα εκδοχή για τη λογική με memristors είναι η αντιμετώπιση της αντίστασης ως λογικής κατάστασης, όπου η υψηλή και η χαμηλή αντίσταση θεωρούνται αντίστοιχα ως τα λογικό μηδέν και άσσος. Για αυτήν την παραδοχή, τα memristors είναι τα βασικά δομικά μέρη της λογικής πύλης. Κάθε memristor λειτουργεί ως μία είσοδος, μία έξοδος και ένα υπολογιστικό λογικό στοιχείο σε διαφορετικά στάδια της υπολογιστικής διαδικασίας [8]. Αυτή η προσέγγιση είναι κατάλληλη για αρχιτεκτονικές κάθετης παράταξης και για αυτό το λόγο μπορεί να χρησιμοποιηθεί σε μία βασική παράταξη memristors η οποία χρησιμοποιείται ευρέως στις εφαρμογές μνήμης. Επίσης αυτή η προσέγγιση είναι ιδιαίτερα ενδιαφέρουσα καθώς μας δίνει την δυνατότητα να εξερευνήσουμε προηγμένες αρχιτεκτονικές υπολογιστών διαφορετικές από τη κλασική αρχιτεκτονική von Neumann. Σε αυτές τις αρχιτεκτονικές, η μνήμη έχει τη δυνατότητα να εκτελεί λογικές λειτουργίες στις ίδιες συσκευές στις οποίες αποθηκεύει δεδομένα, δηλαδή επιτρέπεται η εκτέλεση υπολογισμών μέσα στην μνήμη. Ο υλικός συμπερασμός (material implication) (IMPLY λογική πύλη) είναι ένα παράδειγμα ενός βασικού λογικού στοιχείου το οποίο χρησιμοποιεί αυτή την προσέγγιση, συνδυάζοντας μνήμη κατάστασης με έναν Boolean χειριστή (operator).

##### 3.1.1 Βασική λειτουργία της IMPLY λογικής πύλης.

Η προτεινόμενη memristor-based IMPLY λογική πύλη χρησιμοποιεί έναν αντιστάτη  $R_G$  ( $R_{ON} < R_G < R_{OFF}$ ) συνδεδεμένο με 2 memristors οι οποίοι ονομάζονται P και Q, και παίζουν το ρόλο ψηφιακών διακοπών. Οι αντίστοιχες αρχικές memristances  $p$  και  $q$  είναι οι είσοδοι της πύλης, ενώ η έξοδος της πύλης είναι η τελική memristance του Q. Η βασική ιδέα γύρω από την IMPLY λογική πύλη είναι η εφαρμογή δύο διαφορετικών τάσεων στο P και Q, όπου η  $V_{SET}$ , η εφαρμοσμένη τάση στο Q έχει μεγαλύτερη σημασία από την  $V_{COND}$ , την εφαρμοσμένη τάση στο P ( $|V_{COND}| < |V_{SET}|$ ). Εάν  $p = 1$  (χαμηλή αντίσταση), η τάση στο κοινό τερματικό είναι περίπου  $V_{COND}$  και η τάση στο memristor Q είναι κατά προσέγγιση  $V_{SET} - V_{COND}$ , η οποία είναι σημαντικά μικρή για να διατηρήσει τη λογική κατάσταση του q. Στην περίπτωση όπου  $p = 0$  και  $q = 0$  (υψηλές αντιστάσεις), η εφαρμοζόμενη τάση στο Q είναι περίπου  $V_{SET}$  και το Q είναι στη θέση ON ( $q = 1$ ). Στην περίπτωση κατά την οποία  $p = 0$  και  $q = 1$ , η λογική κατάσταση του q διατηρείται.



Εικόνα 11: Η IMPLY λογική πύλη. Η αρχική κατάσταση του memristor p και q είναι η είσοδος της λογική πύλης και η έξοδος είναι η τελική κατάσταση του memristor q αφού εφαρμοστούν οι τάσεις  $V_{SET}$  και  $V_{COND}$ . Ένας αντιστάτης  $R_G$  είναι συνδεδεμένος και με τα δύο memristors.

### 3.1.2 Ανάλυση της συμπεριφοράς μιας λογικής πύλης

Η  $V_{SET}$  και η  $V_{COND}$ , οι εφαρμοζόμενες τάσεις στο P και Q είναι σταθερές. Για μία οποιαδήποτε αρχική κατάσταση, η κατάσταση q του memristor τείνει να ολισθαίνει προς την κατάσταση ON. Για ψηφιακή λειτουργία, η κατάσταση q θα πρέπει είτε να μείνει αμετάβλητη είτε να αλλάξει και να μετατραπεί σε ON (αλλάζοντας την λογική κατάσταση από λογικό μηδέν σε λογικό ένα).

Οι διαφορετικοί συνδυασμοί υπάρχουν στον πίνακα 1. Εξαιτίας της πόλωσης (polarity) του memristor και των εφαρμοζόμενων τάσεων, η memristance του memristor Q μπορεί μόνο να μειωθεί. Σημειωτέον οι περιπτώσεις 2 και 4, η αρχική λογική κατάσταση του q είναι λογικό ένα και η έξοδος της λογικής πύλης q είναι επίσης λογικό ένα. Με τον τρόπο αυτό λοιπόν η λειτουργία της πύλης ενδυναμώνει την λογική κατάσταση του q αφού η memristance του Q μειώνεται.

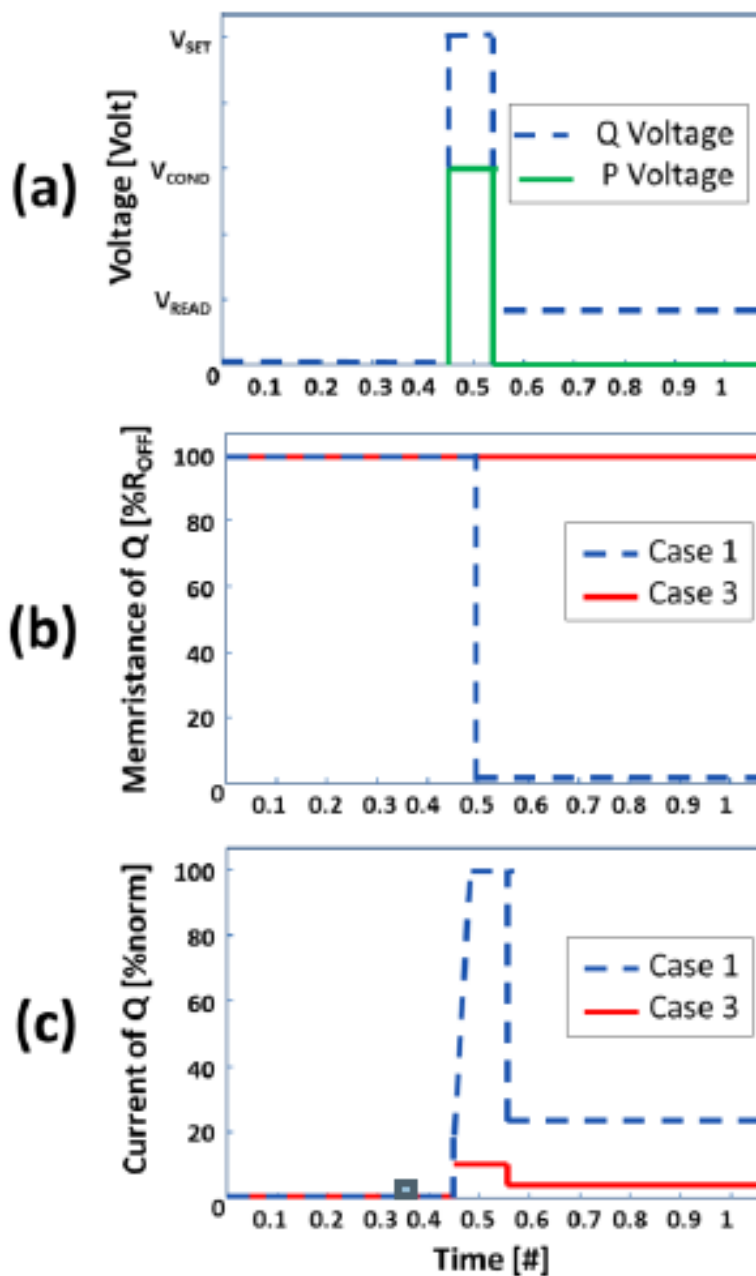


Πίνακας 1: Πίνακας αλήθειας της IMPLY λειτουργίας.

<i>Case</i>	<i>p</i>	<i>q</i>	<i>p<math>\rightarrow</math>q</i>
1	0	0	1
2	0	1	1
3	1	0	0
4	1	1	1

Στην περίπτωση 1, η αρχική κατάσταση του  $q$  είναι λογικό μηδέν. Αφού εφαρμόσουμε τις εξωτερικές τάσεις, το  $q$  αλλάζει σε κατάσταση ON. Αυτή η περίπτωση καθορίζει τον χρόνο που απαιτείται για την εφαρμογή της  $V_{SET}$  και της  $V_{COND}$  μέχρι η λογική κατάσταση του  $q$  να φτάσει στην επιθυμητή κατάσταση (πάνω από ένα συγκεκριμένο επίπεδο αγωγιμότητας το οποίο διατηρεί σωστή λογική συμπεριφορά). Αυτή η περίπτωση καθορίζει το χρόνο εγγραφής του κυκλώματος (το χρόνο καθυστέρησης της λογικής πύλης).

Στην περίπτωση 3, η αρχική κατάσταση του  $q$  είναι το λογικό μηδέν. Αυτή η λογική κατάσταση πρέπει να παραμείνει αμετάβλητη μετά την εφαρμογή της  $V_{SET}$  και της  $V_{COND}$ , αν και οι τάσεις τείνουν να αλλάζουν την αρχική κατάσταση του  $q$  προς την κατάσταση ON του λογικού ένα. Το φαινόμενο αυτό ονομάζεται "ολίσθηση κατάστασης" (state drift). Η κατάσταση λογικού μηδέν του  $q$ , η οποία είναι η έξοδος της πύλης, είναι ηλεκτρολογικά πιο αδύναμη από την κατάσταση λογικού μηδέν του  $q$  ( η memristance του  $Q$  μετά την εφαρμογή των εφαρμοζόμενων είναι χαμηλότερη από την αρχική memristance). Αξίζει να σημειωθεί ότι η ολίσθηση κατάστασης είναι ένα ντετερμινιστικό φαινόμενο.



Εικόνα 12: Συμπεριφορά μίας ιδανικής IMPLY λογικής πύλης. (a) Εφαρμοσμένες τάσεις και στα δύο memristors P και Q. (b) Η Memristance του Q για τις περιπτώσεις 1 και 3. Όσο η memristance στην περίπτωση 1 μειώνεται σε  $R_{ON}$  σε μία περίοδο εγγραφής ενός μηδενικού, η memristance στην περίπτωση 3 δεν αλλάζει. (c) Το ρεύμα του memristor Q. Το ρεύμα στη περίπτωση 1 είναι εμφανώς υψηλό για να μειώσει την αντίσταση του Q.

### 3.1.3 Ταχύτητα της IMPLY λογικής

Η επιτρεπόμενη τιμή του χρόνου που απαιτείται για την εφαρμογή της  $V_{COND}$  και της  $V_{SET}$  καθορίζεται από την περίπτωση 1. Αυτός ο χρόνος εγγραφής είναι ο χρόνος καθυστέρησης

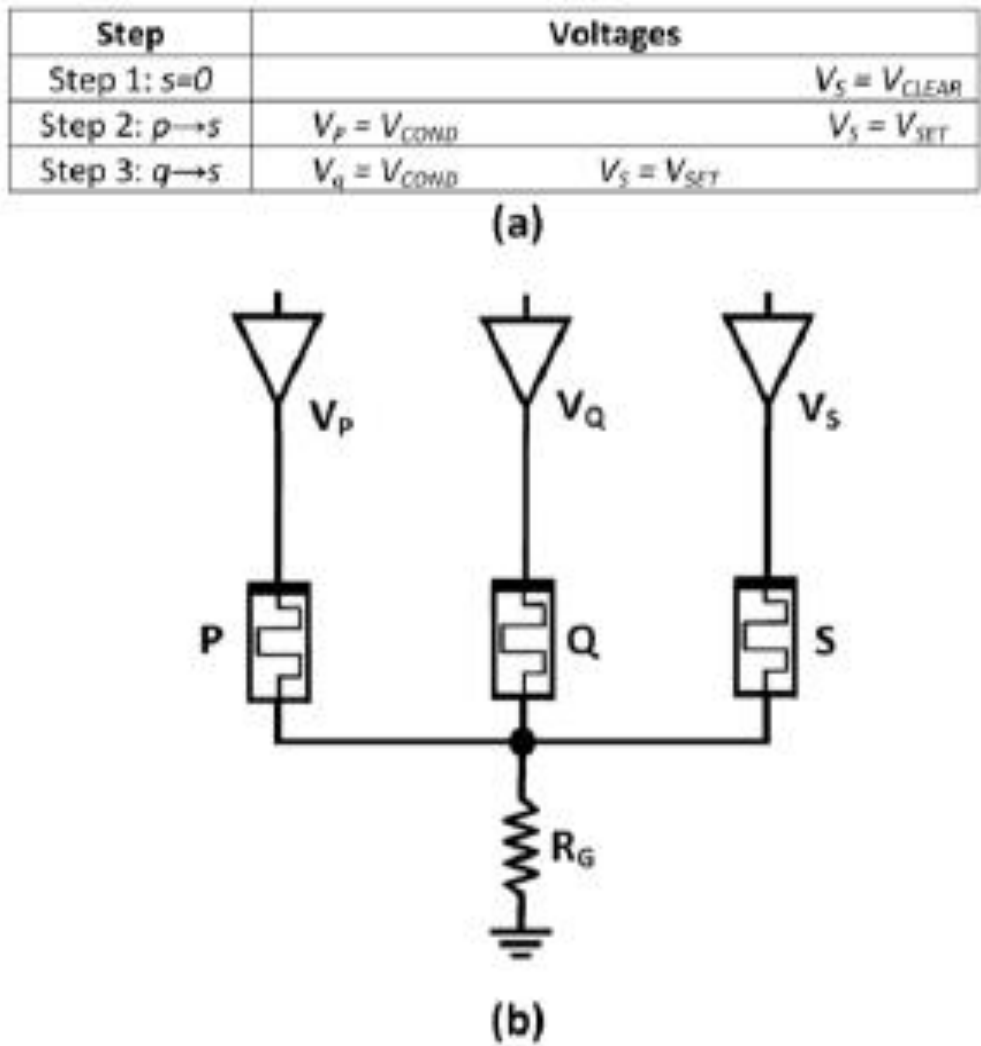
της λογικής πύλης (logic gate) και καθορίζει τις επιδόσεις της λογικής πύλης. Εφόσον η αρχική λογική κατάσταση του memristor είναι άγνωστη κατά τη διάρκεια λειτουργίας, οι τάσεις εφαρμόζονται την ίδια στιγμή για όλες τις περιπτώσεις.

Η ολίσθηση κατάστασης καθορίζεται από την περίπτωση 3, η οποία εξαρτάται από τον χρόνο εγγραφής που καθορίζεται για την περίπτωση 1. Επιπλέον οποιαδήποτε βελτίωση στις επιδόσεις λόγω αλλαγών στις εφαρμοζόμενη τάση αυξάνει την ολίσθηση κατάστασης και μειώνει την ευρωστία της λογικής πύλης [9].

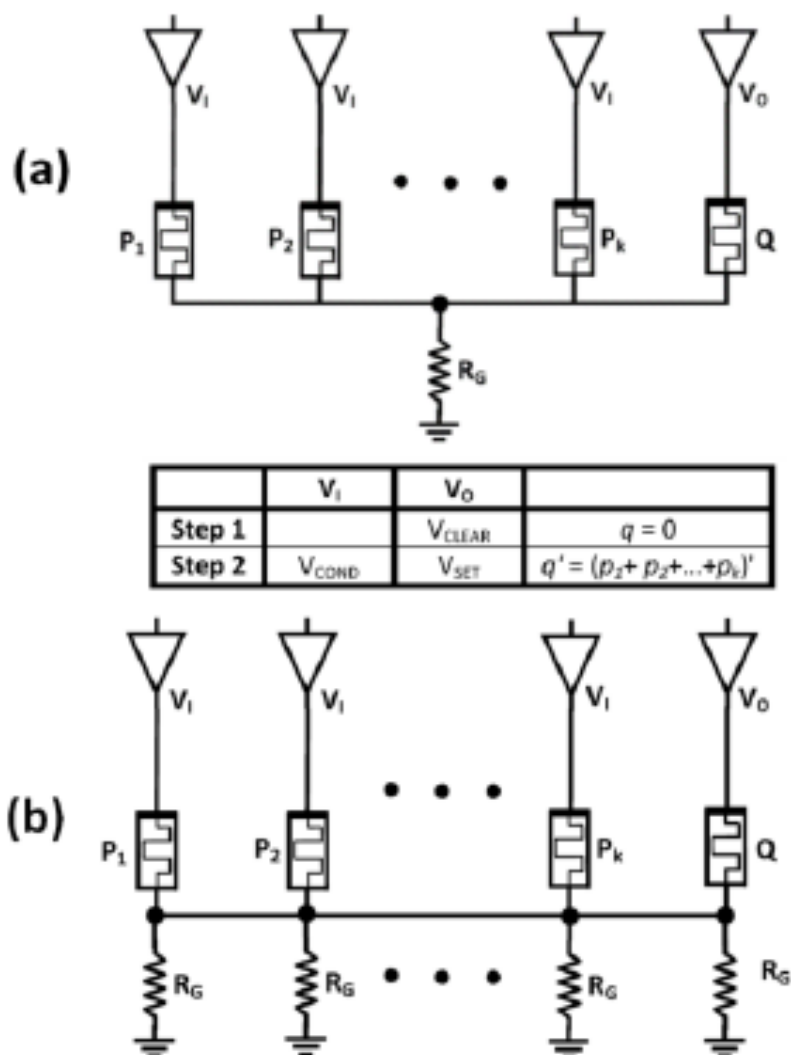
### 3.1.4 Εκτεταμένες λογικές λειτουργίες βασισμένες στην IMPLY

Οποιαδήποτε γενική Boolean λογική  $f: B^n \rightarrow B$  είναι σε θέση να κατασκευαστεί με την βοήθεια  $n + 3$  memristors [10], όπου 3 επιπλέον memristors αναλαμβάνουν τον υπολογισμό. Μόνο 2 memristors απαιτούνται για μέχρι και 3 εισόδους. Ο υπολογισμός της λειτουργίας γίνεται σε βήματα. Σε κάθε βήμα είτε εφαρμόζεται μια ψευδής λογική είτε εφαρμόζεται μία IMPLY λογική σε δύο memristors, όπου το αποτέλεσμα γράφεται σε ένα memristor (το οποίο είναι μία από τις εισόδους του σταδίου υπολογισμού της IMPLY). Αυτή η διαδικασία απαιτεί μία μακρά αλληλουχία λειτουργιών ανάλογα με τον αριθμό των εισόδων. Αυτή η μεθοδολογία έχει βελτιωθεί με την προσθήκη μόνο δύο επιπλέον memristors αντί τριών [11]. Ενώ έχει αναπτυχθεί ένας γενικός αλγόριθμος για τον υπολογισμό οποιασδήποτε Boolean λειτουργίας με έναν πολύ μικρό αριθμό memristors, η υπολογιστική διαδικασία απαιτεί ένα πολύ μεγάλο αριθμό από στάδια λειτουργίας και για αυτό το λόγο απαιτεί και σημαντικό υπολογιστικό χρόνο. Το σχηματικό και η ακολουθία μίας NAND δύο εισόδων βασισμένης σε IMPLY πύλη και μία ψευδής λογική πύλη απεικονίζονται στην εικόνα 13. Αυτή η NAND πύλη έχει σχεδιαστεί να ελαχιστοποιεί τον υπολογιστικό χρόνο και ο αριθμός των memristors έχει μειωθεί στα τρία. Η λειτουργία αυτής της NAND λογικής πύλης αλλάζει την λειτουργία του κάθε memristor κατά τη διάρκεια της υπολογιστικής διαδικασίας. Δύο memristors λειτουργούν ως είσοδοι στο αρχικό στάδιο, ένα memristor λειτουργεί ως το αποτέλεσμα στο τελικό στάδιο, και όλα τα memristor λειτουργούν μαζί ως ένα υπολογιστικό λογικό στοιχείο (ως μία βασισμένη σε memristor IMPLY πύλη) κατά τη διάρκεια των διαφορετικών σταδίων της υπολογιστικής διαδικασίας. Αυτή η εφαρμογή απαιτεί τρία υπολογιστικά στάδια (computational stages) ( ένα ψευδές και δύο IMPLY).

Η IMPLY λογική πύλη μπορεί επίσης να επεκταθεί σε μια NOR λογική πύλη πολλαπλών εισόδων. Η λειτουργία αυτής της NOR πύλης απαιτεί δύο υπολογιστικά στάδια, το πρώτο στάδιο αρχικοποιεί το Q σε λογικό μηδέν ( $q = 0$ ) και το δεύτερο στάδιο εφαρμόζει τη  $V_{SET}$  και τη  $V_{COND}$  με ένα τρόπο παρόμοιο με την κοινή IMPLY λογική.



Εικόνα 13: IMPLY NAND λογική πύλη. (a) Λογική λειτουργία μίας βασισμένης σε IMPLY NAND, η λογική πύλη απαιτεί 3 διαδοχικά βήματα και (b) σχηματικό μίας βασισμένης σε IMPLY NAND πύλης.



Εικόνα 14: Μία επέκταση σε μια IMPLY κ-εισόδων NOR. (a) Σχηματικό βασισμένο σε μία εκτέλεση πολλαπλών λειτουργιών σε ένα μόνο βήμα, και (b) μία βελτιωμένη κατασκευή όπου οι αντιστάτες φόρτωσης είναι αφιερωμένοι στις συμμετέχοντες λογικές συσκευές.

### 3.1.5 IMPLY μέσα σε memristor-based crossbar

Η IMPLY λογική πύλη δεν μπορεί εύκολα να συνεργαστεί με τη βασική CMOS λογική αφού οι κατασκευή και των δύο κυκλωμάτων διαφέρει σημαντικά. Στην IMPLY λογική οικογένεια, μία αντίσταση αντί για μία τάση αναπαριστά τη λογική κατάσταση. Επιπλέον, για τη λειτουργία της λογικής πύλης, μία αλληλουχία συγκεκριμένων τάσεων εφαρμόζεται στα memristors. Η IMPLY λογική πύλη για αυτό το λόγο απαιτεί πολλά υπολογιστικά στάδια (συνήθως ένα υπολογιστικό στάδιο εκτελείται κατά τη διάρκεια διαφορετικού κύκλου ρολογιού (clock cycle)), και έναν ξεχωριστό μηχανισμό για το διάβασμα του αποτελέσματος του υπολογισμού και τον χειρισμό των τάσεων. Για τη συνεργασία της IMPLY λογικής πύλης με τη βασική βασισμένη σε τάση CMOS λογική, απαιτείται ένας μηχανισμός μετατροπής. Αυτός ο μηχανισμός περιλαμβάνει έναν ενισχυτή καθώς και άλλα επιπλέον εξαρτήματα. Το επιπλέον κύκλωμα μειώνει την αποδοτικότητα της συνεργασίας της CMOS με μία IMPLY λογική πύλη.

Εναλλακτικά, η IMPLY λογική πύλη μπορεί να ενσωματωθεί μέσα σε συστοιχία βασισμένη σε

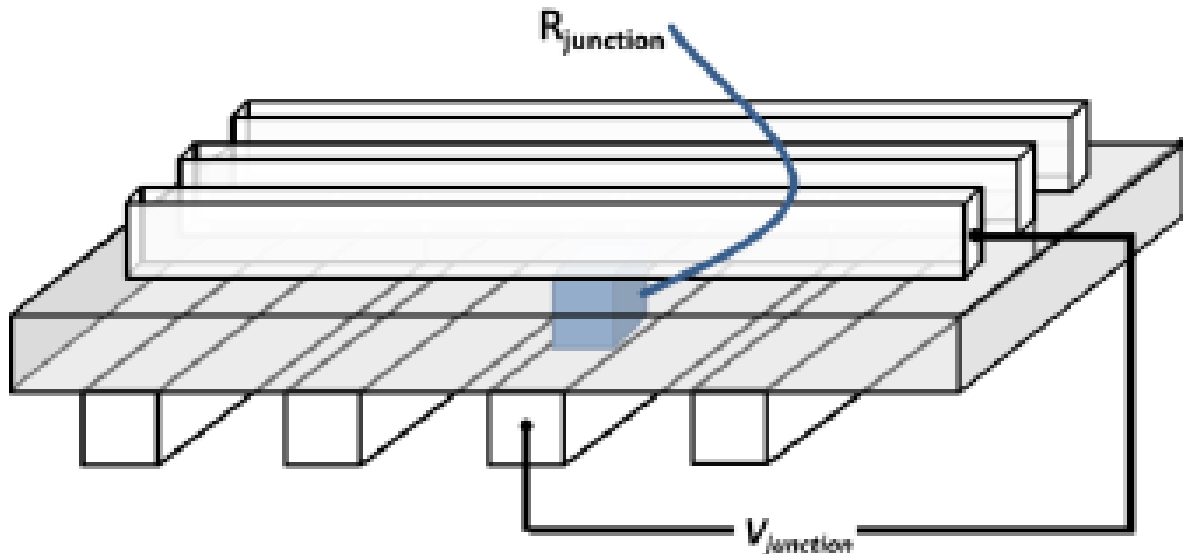
memristors η οποία θα χρησιμοποιείται για μνήμη, όπου οι τιμές εισόδου και εξόδου αποθηκεύονται σε κελιά μνήμης (memory cells). Αυτή η ενσωμάτωση μειώνει την ενέργεια η οποία καταναλώνεται και δίνει μια ευκαιρία σε όλες τις νέες μη-von Neumann αρχιτεκτονικές.

### 3.1.6 Memristor-based crossbar

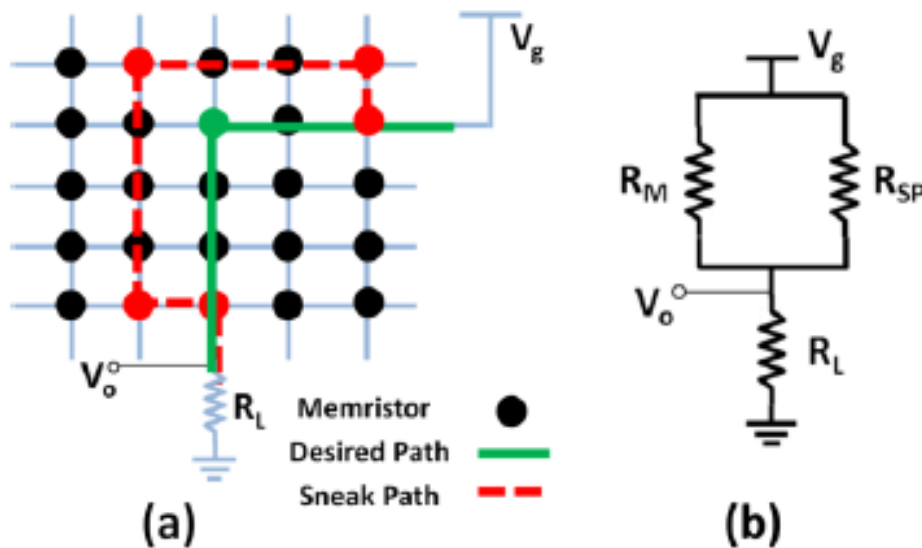
Η βασική δομή (structure) ενός memristor-based crossbar αποτελείται από δύο σετ παράλληλων αγωγίμων μεταλλικών γραμμών. Οι αγωγιμες γραμμές είναι κάθετες και συμπεριφέρονται ως άνω και κάτω ηλεκτρόδια στο memristive υλικό το οποίο βρίσκεται ανάμεσα στις γραμμές [12]. Η βασική δομή ενός memristor-based crossbar απεικονίζεται στην εικόνα 15. Η λειτουργία εγγραφής σε ένα κελί μέσα στο crossbar επιτυγχάνεται εφαρμόζοντας μια συγκεκριμένη τάση στον κόμβο, όπου η τάση εφαρμόζεται και στις δύο γραμμές. Για παράδειγμα, για την εγγραφή ενός λογικού άσσου (χαμηλή αντίσταση), μία θετική τάση εφαρμόζεται στην γραμμή στήλης και η γείωση στην γραμμή σειράς (μία θετική τάση εφαρμόζεται στο memristor). Για την εγγραφή ενός λογικού μηδέν (υψηλή αντίσταση), η γραμμή στήλης συνδέεται με γείωση και μία θετική τάση συνδέεται στη γραμμή σειράς (μία αρνητική τάση εφαρμόζεται στο memristor). Αυτές οι τάσεις ονομάζονται μερικές φορές  $V_{SET}$  (θετική τάση για την εγγραφή ενός λογικού άσσου, όχι απαραίτητα όπως στην IMPLY) και  $V_{RESET}$  (αρνητική τάση για την εγγραφή ενός λογικού μηδέν). Αφού τα memristor έχουν την δυνατότητα μόνιμης αποθήκευσης, τα δεδομένα δεν αλλάζουν όταν δεν εφαρμόζεται καμία τάση στις γραμμές. Η δομή του crossbar επιτρέπει η πυκνότητα της μνήμης να είναι πολύ υψηλή λόγω του ότι τα CMOS transistor δεν χρησιμοποιούνται για το κάθε κελί μνήμη, αλλά μόνο για την επιλογή της γραμμής. Αυτή η δομή μνήμης είναι περισσότερες από είκοσι φορές πιο πυκνή από την DRAM.

Η λειτουργία διαβάσματος του crossbar επιτυγχάνεται από εφαρμόζοντας μία σχετικά μικρή τάση (μικρότερη από  $V_{SET}$ ) σε ένα κόμβο και μετρώντας την τιμή του ρεύματος. Από το νόμο του Ohm, η αντίσταση του memristor καθορίζεται από αυτή τη τιμή του μετρημένου ρεύματος. Η μέτρηση του ρεύματος συνήθως επιτυγχάνεται μετατρέποντας το ρεύμα σε μία τάση μέσω ενός διαιρέτη τάσης με μία γνωστή αντίσταση  $R_{pu}$ . Η τάση αυτή  $v_s$  συγκρίνεται με μία γνωστή τάση.

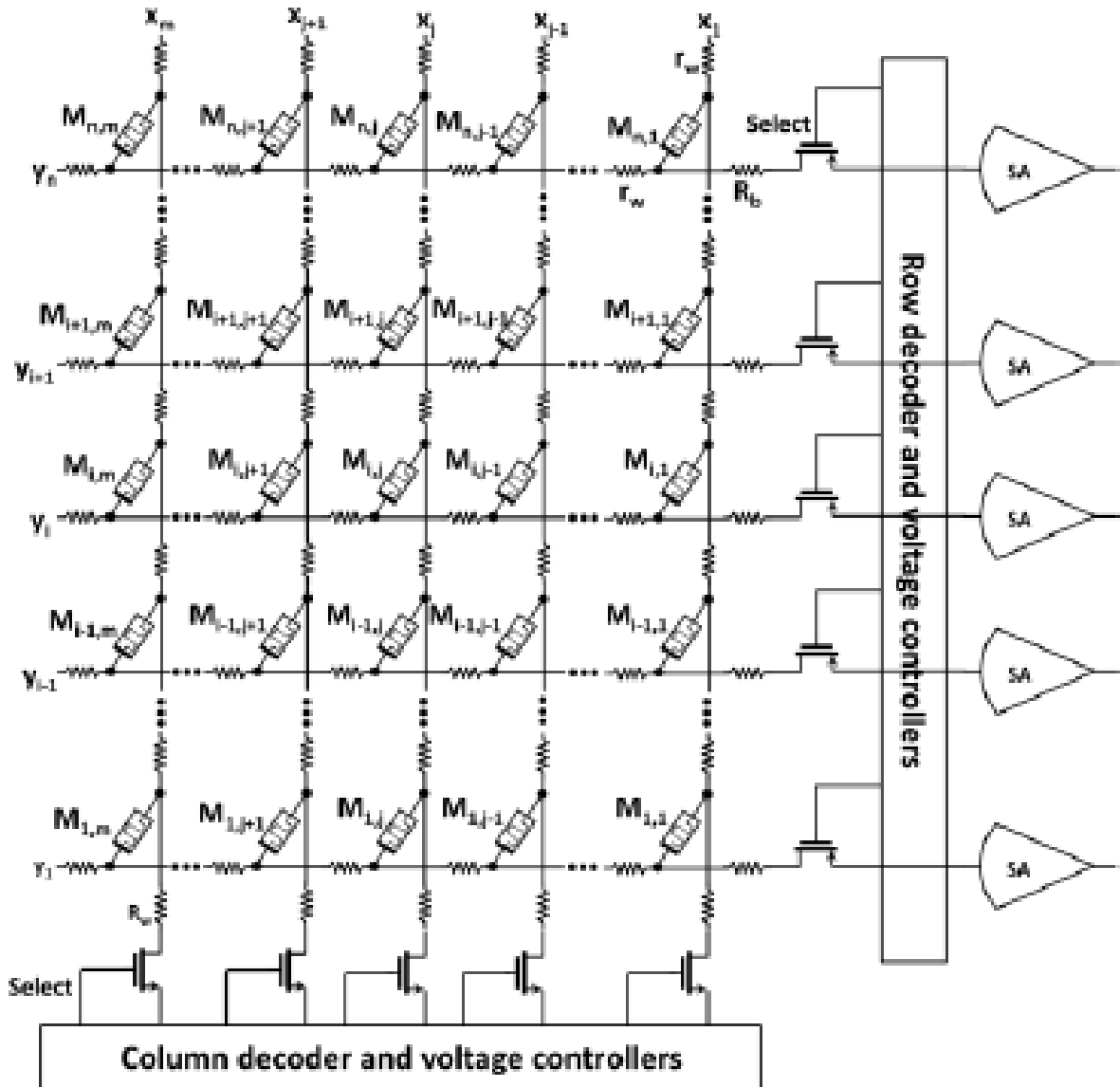
Ένα ανεπιθύμητο φαινόμενο στα crossbar είναι αυτό των sneak paths [13], τα οποία είναι ανεπιθύμητα μονοπάτια για τη ροή του ρεύματος. Όταν μία τάση εφαρμόζεται σε έναν κόμβο στο crossbar, το ρεύμα επίσης ρέει και σε μονοπάτια διαφορετικά από το επιθυμητό. Αυτά τα μονοπάτια διαπερνούν περισσότερα από ένα memristors και προσθέτουν μία αντίσταση παράλληλα στην αντίσταση του memristor στον κόμβο ο οποίος διαβάζεται. Μία απεικόνιση του φαινομένου sneak paths υπάρχει στην εικόνα 16. Αυτή η παράλληλη αντίσταση εξαρτάται από τα αποθηκευμένα δεδομένα στα memristors τα οποία είναι στα ανεπιθύμητα μονοπάτια και αλλάζει τη τάση  $v_s$  από έναν απλό διαιρέτη τάσης ανάμεσα στη  $R_{pu}$  και στην αντίσταση ενός memristor σε έναν διαιρέτη τάσης ανάμεσα στην  $R_{pu}$  και στη συνολική αντίσταση όλων των memristors σε όλα τα μονοπάτια. Ένα σχηματικό του crossbar συμπεριλαμβανομένου του μηχανισμού εγγραφής και διαβάσματος απεικονίζεται στην εικόνα 17. Υπάρχουν αρκετές προσεγγίσεις για την εξάλειψη ή τη μείωση των snack paths όπως για παράδειγμα η γείωση μη ενεργών γραμμών.



Εικόνα 15 : Βασική δομή ενός memristor-based crossbar. Κάθε κόμβος στις παράλληλες γραμμές είναι ένα κελί μνήμης με μεταβλητή αντίσταση .

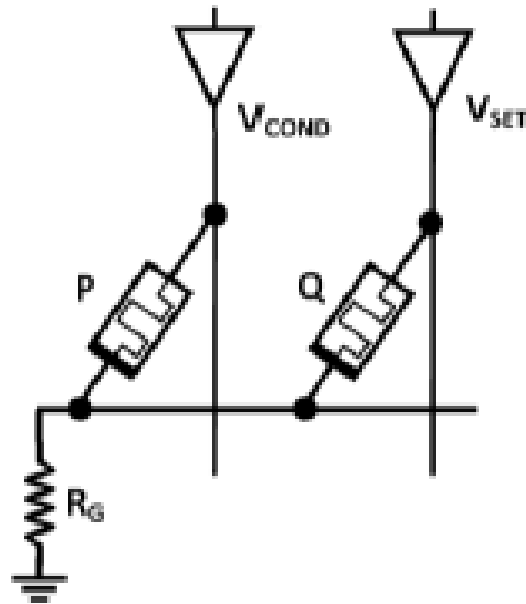


Εικόνα 16: Sneak paths σε ένα memristive crossbar. (a) Ένα παράδειγμα sneak path. Το επιθυμητό μονοπάτι έχει σχεδιαστεί με συνεχόμενη γραμμή ενώ το sneak path με διακεκομμένη. (b) Το αντίστοιχο κύκλωμα. Όλα τα sneak paths έχουν μία ισοδύναμη αντίσταση συνδεδεμένη παράλληλα με αυτή του memristor.



Εικόνα 17:  $m \times n$  memristive crossbar.



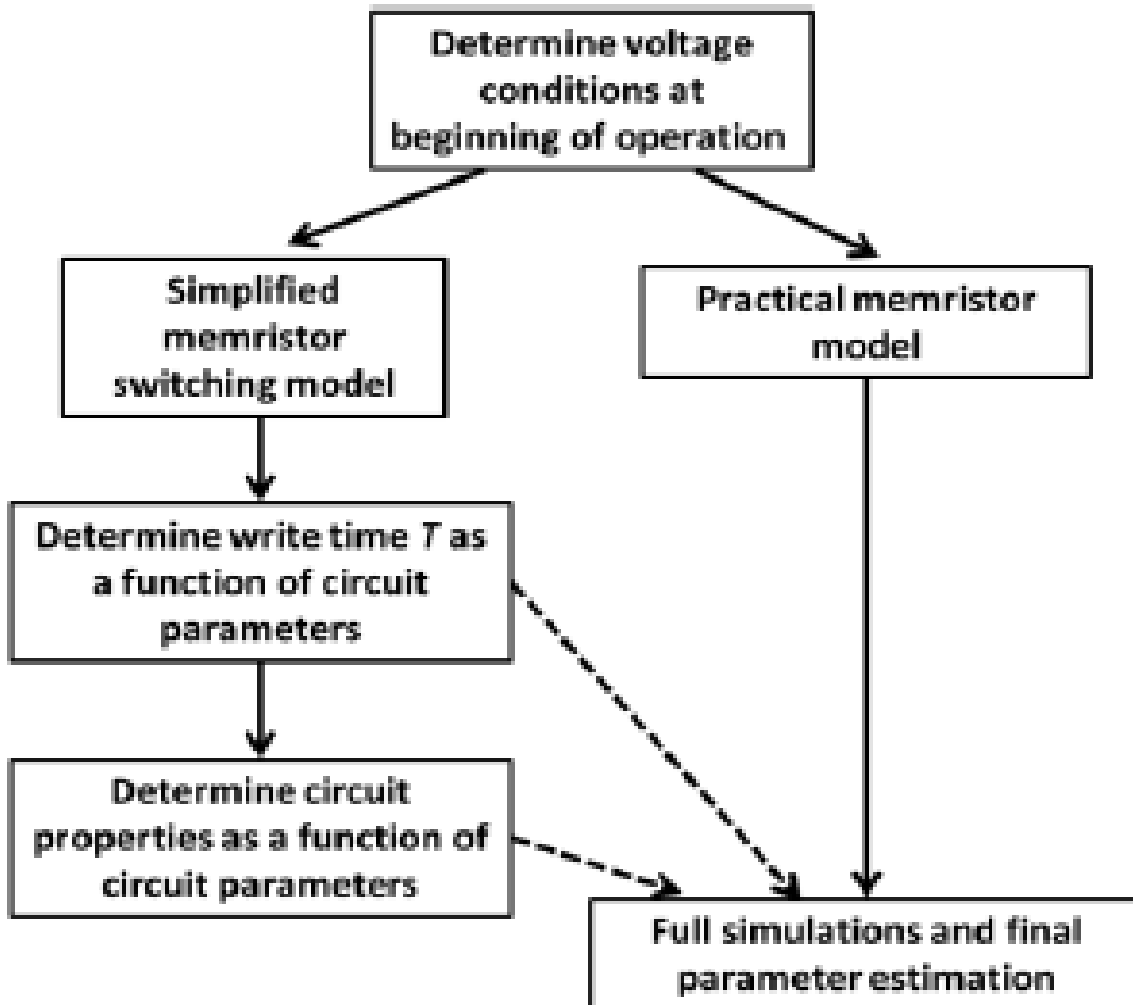


Εικόνα 18: Μία IMPLY λογική πύλη μέσα σε ένα memristor-based crossbar.

### 3.1.7 Μεθοδολογία σχεδιασμού λογικών πυλών

Αν και κανένα ολοκληρωμένο και έγκυρο μοντέλο δεν υπάρχει ακόμα, όλα τα προτεινόμενα μοντέλα είναι ακόμη αρκετά πολύπλοκα και η ακριβής συμπεριφορά της memristive λογικής είναι μαθηματικά δυσπρόσιτη. Για το λόγο αυτό υπάρχει η ανάγκη για heuristics για το σχεδιασμό memristive κυκλωμάτων. Για τις memristive-based IMPLY λογικές πύλες, οι κατάλληλες παράμετροι κυκλώματος ( $R_G$ ,  $V_{SET}$ ,  $V_{COND}$ , και ο χρόνος εφαρμογής των τάσεων  $T$ ) χρειάζεται να καθοριστούν κάτω από μερικούς γενικούς περιορισμούς. Αυτοί οι περιορισμοί συμπεριλαμβάνουν ελαχιστοποίηση της κατανάλωσης ενέργειας (μόνο δυναμικής ισχύος σε ένα memristor-based crossbar), μείωση του χώρου (ο αριθμός των ενεργών memristors σε ένα crossbar και ο αριθμός των transistors σε ένα χειριστή), μείωση του χρόνου καθυστέρησης της λογικής πύλης και αύξηση της ευρωστίας του κυκλώματος (μειώνοντας την ολίσθηση αντίστασης κατά τη διάρκεια της λειτουργίας για εκείνες τις εισόδους για τις οποίες το λογικό αποτέλεσμα δεν αλλάζει). Η παρασιτική χωρητικότητα των CMOS transistors τα οποία είναι συνδεδεμένα με το crossbar και η παρασιτική αντίσταση των μεταλλικών γραμμών καθώς και το φαινόμενο sneak path πρέπει επίσης να ληφθούν υπόψη.

Μία γενική ροή για τον σχεδιασμό μιας memristor-based IMPLY λογικής πύλης απεικονίζεται στη εικόνα 19. Μετά τον καθορισμό της τοπολογίας του κυκλώματος, οι συνθήκες κατά το ξεκίνημα της λειτουργίας πρέπει να επίσης να καθοριστούν. Αυτές οι στατικές συνθήκες δεν βασίζονται στο μοντέλο του memristor και είναι αναγκαίες για τη σωστή συμπεριφορά του κυκλώματος. Τα απλοποιημένα μοντέλα memristors χρησιμοποιούν πολλά heuristics για την προσέγγιση των χαρακτηριστικών του κυκλώματος. Το μοντέλο TEAM [14] χρησιμοποιείται εδώ για να εκτιμηθούν οι παράμετροι του κυκλώματος.



Εικόνα 19: Σχεδιαστική ροή για memristor-based IMPLY λογικές πύλες.

Στο σχέδιο μίας βασικής IMPLY λογικής πύλης, οι παράμετροι κυκλώματος  $V_{SET}$ ,  $V_{COND}$ ,  $R_G$  καθώς και ο χρόνος εφαρμογής των τάσεων  $T$  πρέπει να καθοριστούν. Οι παράμετροι του memristor είναι προκαθορισμένοι για μία δεδομένη τεχνολογία.

Αν και είναι δύσκολος ο υπολογισμός της εξέλιξης του χρόνου της τάσης στο Q (εικόνα 4), είναι πιθανός ο προσδιορισμός της τάσης στο Q στην αρχή της δραστηριότητας της λογικής πύλης. Η αρχική εφαρμοζόμενη τάση στο Q είναι διαφορετική για κάθε περίπτωση (διαφορετική αρχική memristance για Q και P). Οι αρχικές τάσεις υπάρχουν στον πίνακα 2 υπό την προϋπόθεση ότι η memristance του λογικού ένα και του λογικού μηδέν είναι  $R_{ON}$  και  $R_{OFF}$ , όπου  $R_{OFF} \gg R_{ON}$ .

Από τις αρχικές εφαρμοζόμενες τάσεις, μερικές απαραίτητες συνθήκες για σωστή λογική συμπεριφορά μπορούν να καθοριστούν. Η βασική σχεδιαστική αρχή είναι ότι ο χρόνος εγγραφής της λογικής πύλης καθορίζεται από την περίπτωση εισόδου 1 (βλέπε πίνακα 2), αλλά το κύκλωμα δεν θα πρέπει να υπερβαίνει μια συγκεκριμένη ολίσθηση τάσης στην περίπτωση εισόδου 3.

Ένα πολύ χρήσιμο μοντέλο είναι το μοντέλο της δυαδικής memristance [15]. Υποθέτουμε ότι επιτρέπονται μόνο δύο memristances,  $R_{ON}$  και  $R_{OFF}$ . Ένα συνολικό φορτίο  $Q'$  ρέει διαμέσου του memristor για να προκαλέσει τη memristance  $R_{OFF}$  να αλλάξει σε memristance  $R_{ON}$ . Με αυτές τις υποθέσεις αλλά και λύνοντας το πρόβλημα της συμπεριφοράς εναλλαγής στην περίπτωση 1 και του χρόνου εγγραφής  $T$  ως λειτουργίας του  $Q'$ , η παράμετρος του κυκλώματος  $T$  είναι:

$$T = \left[ \frac{R_{OFF}^2 + 2R_{OFF}R_G}{R_{OFF}V_{SET} + R_G[V_{SET} - V_{COND}]} \right] \cdot Q'$$

Ο χρόνος εγγραφής για διαφορετικές παραμέτρους κυκλώματος και διαφέρουσα  $V_{SET}$  απεικονίζεται στην εικόνα 20. Αξίζει να σημειωθεί ότι η λογική πύλη είναι πιο γρήγορη με μία υψηλότερη εφαρμοζόμενη τάση η μικρότερη  $R_{OFF}$ .

Με αυτό το μοντέλο υπόψη είναι πιθανή η οριοθέτηση της ολίσθησης κατάστασης (περίπτωση 3 στον πίνακα 2) για μία προκαθορισμένη ολίσθηση. Η ολίσθηση κατάστασης είναι:

$$q_q(T) \approx \left[ V_{SET} - \frac{R_G}{R_{ON} + R_G} V_{COND} \right] \cdot \left[ \frac{R_{OFF} + 2R_G}{R_{OFF}V_{SET} + R_G[V_{SET} - V_{COND}]} \right] \cdot Q'$$

Όπου  $q_q(T)$  είναι το συνολικό φορτίο το οποίο ρέει διαμέσου του memristor  $Q$  μετά από χρόνο  $T$ , όπως στην περίπτωση 3. Εάν η ολίσθηση κατάστασης είναι οριοθετημένη σε μία κατάσταση  $Q'/4$  ως η μέγιστη ολίσθηση κατάστασης, μετά από τέσσερις εκτελέσεις της λογικής πύλης στην περίπτωση 3 η ολίσθηση κατάστασης θα αλλάξει τη memristive λογική κατάσταση της  $q$ . Αυτό το φαινόμενο απαιτεί μία ανανέωση κάθε τρεις εκτελέσεις της λογικής πύλης αφού η λογική κατάσταση θα άλλαζε στην ανεστραμμένη τιμή κατά τη διάρκεια της τέταρτης εκτέλεσης. Η επιτρεπόμενη τιμή της  $V_{SET}$  για πολλές παραμέτρους κυκλωμάτων απεικονίζεται στην εικόνα 21. Αξίζει να σημειωθεί ότι η ολίσθηση κατάστασης είναι πιο μεγάλη όταν με μία υψηλότερη εφαρμοζόμενη τάση, η με μία μικρότερη  $R_{OFF}$ . Συνδυάζοντας τις εικόνες 20 και 21 βγάζουμε το συμπέρασμα για την ταχύτητα και την ευρωστία της memristive IMPLY λογικής πύλης το οποίο απεικονίζεται στην εικόνα 22.

Ένα ακόμα απλό και ιδιαίτερα χρήσιμο μοντέλο memristor ακολουθεί μη γραμμική συμπεριφορά με προκαθορισμένη τάση κατωφλίου  $V_{ON}$ . Υπό αυτό το μοντέλο, για μία εφαρμοζόμενη τάση κάτω από  $V_{ON}$ , η memristance παραμένει αμετάβλητη. Για την παραγωγή σωστής λογικής συμπεριφοράς, η αρχική εφαρμοζόμενη τάση στο  $Q$  πρέπει να είναι πάνω από τη τάση κατωφλίου στην περίπτωση 1 και κάτω από τάση κατωφλίου στην περίπτωση 3. Προσθέτοντας αυτή την υπόθεση στην αρχική εφαρμοζόμενη τάση (βλέπε πίνακα 2) οδηγούμαστε στις δύο επόμενες συνθήκες:

Εφαρμογές memristors σε VLSI κυκλώματα

$$R_{ON} \cdot \frac{V_{SET} - V_{ON}}{V_{ON} - [V_{SET} - V_{COND}]} < R_G < R_{OFF} \cdot \frac{V_{SET} - V_{ON}}{2V_{ON} - [V_{SET} - V_{COND}]},$$

$$\frac{V_{SET}}{V_{COND}} < \frac{R_{OFF}}{R_{ON}}.$$

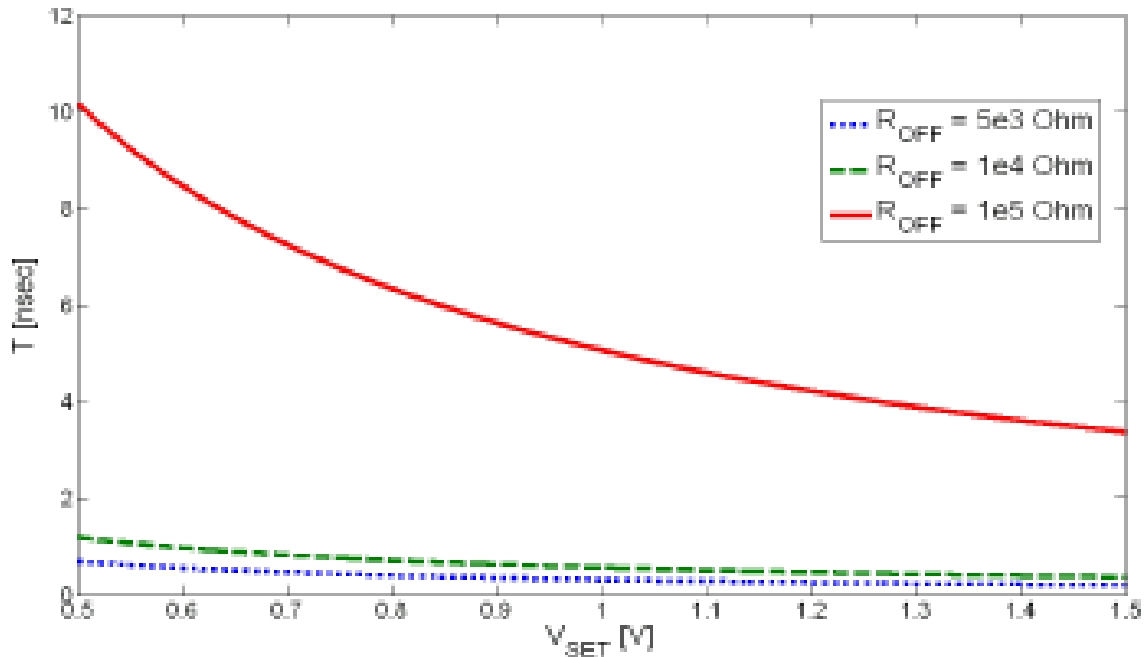
Η επιτρεπόμενη τιμή για το  $R_G$  για διάφορες παραμέτρους του κυκλώματος με διαφέρουσα  $V_{SET}$  απεικονίζεται στην εικόνα 23. Μία λογική τιμή για το  $R_G$  είναι η γεωμετρική μέση των  $R_{ON}$  και  $R_{OFF}$ ,

$$R_G = \sqrt{R_{ON} \cdot R_{OFF}},$$

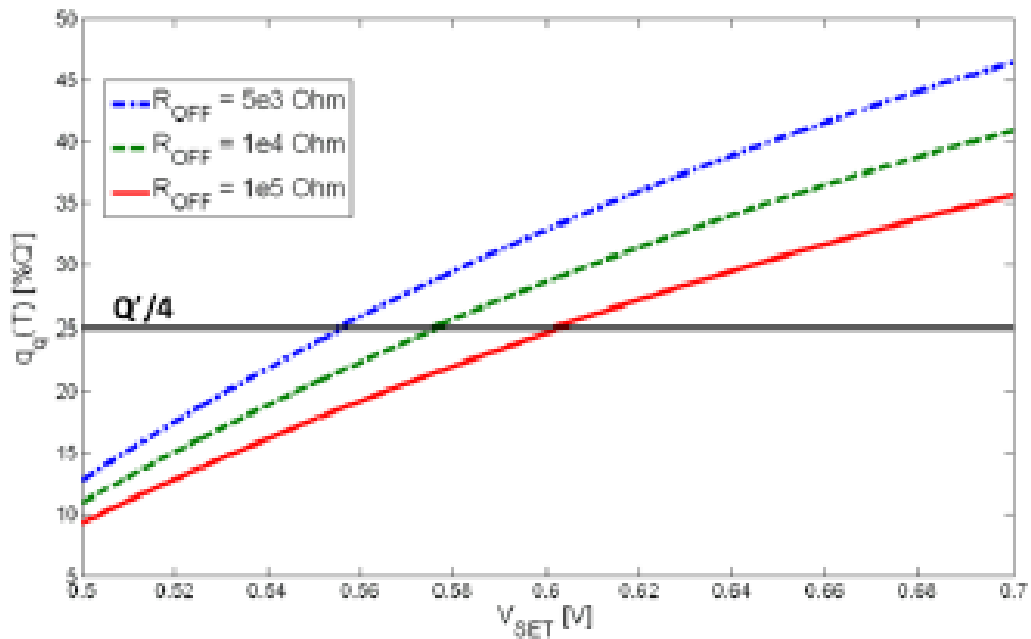
έτσι ώστε να διατηρηθεί η σταθερή αναλογία (ratio) ανάμεσα σε κάθε ζευγάρι αντιστάσεων,  $R_{ON}$  και  $R_{OFF}$ ,  $R_G$  και  $R_{OFF}$ . Άλλες τιμές της  $R_G$  είναι επίσης πιθανές.

Πίνακας 2: Τάσεις εισόδου πύλης  $V_Q$  και  $V_P$ , αντίστοιχα στα memristors P και Q  $Q_{SET} = 0$ , υπό την υπόθεση ότι η memristance του λογικού ένα και μηδέν είναι  $R_{ON}$  και  $R_{OFF}$ , όπου  $R_{OFF} \gg R_{ON}$ .

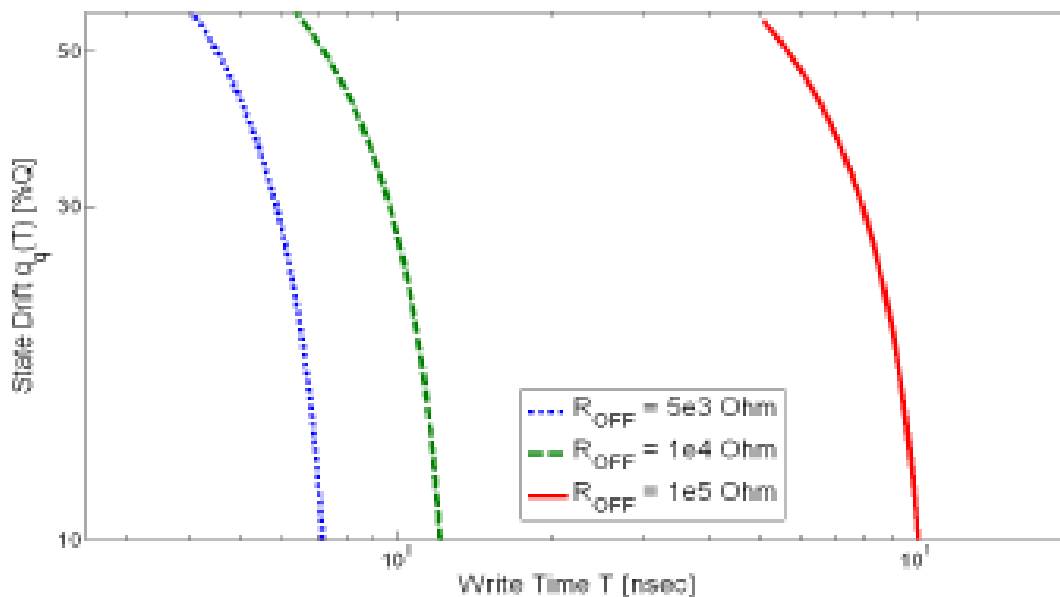
Case	$V_Q(t=0)$	$V_P(t=0)$
1	$\frac{R_{OFF} + R_G}{R_{OFF} + 2R_G} \cdot V_{SET} - \frac{R_G}{R_{OFF} + 2R_G} \cdot V_{COND}$	$-\left[ \frac{R_G}{R_{OFF} + 2R_G} \cdot V_{SET} - \frac{R_{OFF} + R_G}{R_{OFF} + 2R_G} \cdot V_{COND} \right]$
2	$V_{SET} \cdot \frac{R_{ON}}{R_{OFF}} \cdot \frac{R_{OFF} + R_G}{R_{ON} + R_G} \approx V_{SET}$	$-\left[ V_{SET} \cdot \frac{R_G}{R_{ON} + R_G} - V_{COND} \right]$
3	$V_{SET} - V_{COND} \cdot \frac{R_G}{R_{ON} + R_G}$	$V_{COND}$
4	$V_{SET} \cdot \frac{R_{ON} + R_G}{R_{ON} + 2R_G} - V_{COND} \cdot \frac{R_G}{R_{ON} + 2R_G}$	$-\left[ V_{SET} \cdot \frac{R_G}{R_{ON} + 2R_G} - V_{COND} \cdot \frac{R_{ON} + R_G}{R_{ON} + 2R_G} \right]$



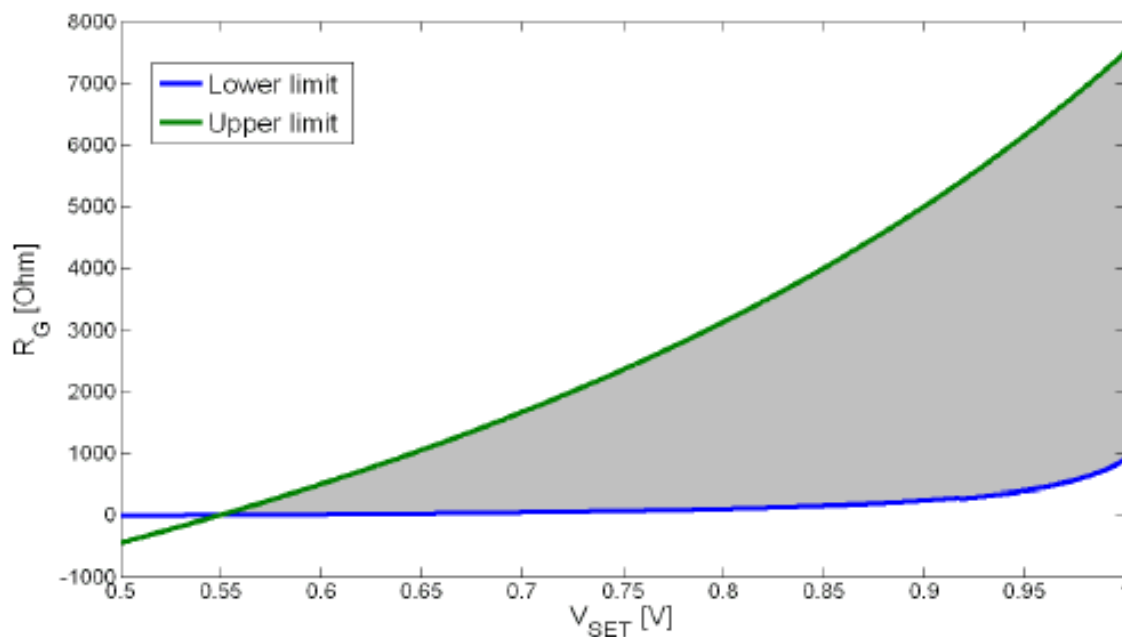
Εικόνα 20: Επιτρεπόμενος χρόνος εγγραφής  $T$  στην περίπτωση 1 για τρεις τιμές της  $R_{OFF}$  (5 kΩ, 10kΩ, και 100 kΩ) υπό την υπόθεση του μοντέλου δυαδικής αντίστασης και  $Q' = 5 * 10^{-14} \text{ C}$ .



Εικόνα 21: Επιτρεπόμενες τιμές της  $V_{SET}$  για οριοθετημένη ολίσθηση τάσης στην περίπτωση 3 του  $Q'/4$ . Η  $V_{SET}$  επιτρέπεται εάν  $q_q(T)$  είναι μικρότερη από  $Q'/4$  (η οριζόντια στην εικόνα)



Εικόνα 22: Tradeoff ανάμεσα στην ταχύτητα (χρόνος εγγραφής) και στην ευρωστία (την ολίσθηση κατάστασης στην περίπτωση 3 για το memristor Q) για τρεις τιμές της  $R_{OFF}$  (5 k $\Omega$ , 10k $\Omega$ , και 100 k $\Omega$ ) υπό την υπόθεση του μοντέλου δυαδικής αντίστασης και  $Q' = 5 \times 10^{-14}$  C.



Εικόνα 23: Η επιτρεπόμενη τιμή της  $R_G$  εξαρτάται από την  $V_{SET}$ . Η πάνω γραμμή είναι το άνω όριο για την επιτρεπόμενη  $R_G$  και η κάτω γραμμή είναι χαμηλότερο όριο για την  $R_G$ . Υπό αυτήν την υπόθεση της τάσης κατωφλίου  $V_{ON} = 0.55$  V,  $V_{COND} = 0.5$  V,  $R_{ON} = 100$   $\Omega$  και  $R_{OFF} = 10$  k $\Omega$ .

### 3.1.8 Παράδειγμα IMPLY λογικής πύλης ενός bit

Ας εφαρμόσουμε ένα συγκεκριμένο παράδειγμα της εφαρμογής του διαγράμματος ροής της εικόνας 19, με την υπόθεση ότι ο μέγιστος χρόνος εγγραφής είναι 0.5  $\mu\text{sec}$ . Σημειωτέον ότι ο πραγματικός χρόνος εγγραφής ενός memristor είναι σημαντικά γρηγορότερος [16]. Η μέγιστη επιτρεπόμενη ολίσθηση κατάστασης είναι  $0.00001R_{\text{OFF}}$  (0.001% της ολίσθησης κατάστασης αν συγκριθεί με ολική αλλαγή, ισάξια με  $10^5$  εκτελέσεις της λογικής πύλης πριν την ολική εναλλαγή).

Ας υποθέσουμε ότι ένα memristor με  $R_{\text{ON}}$  και  $R_{\text{OFF}}$ , αντίστοιχα με αντίσταση 1 k $\Omega$  και 100k $\Omega$ . Θέτουμε μία παράμετρο κυκλώματος  $V_{\text{COND}}$  στα 0.5 Volts. Από τις εικόνες 20 και 21, παρατηρείστε ότι η  $V_{\text{SET}}$  αυξάνεται, ο χρόνος εγγραφής της λογικής πύλης  $T$  μειώνεται και η απόκριση της πύλης είναι γρηγορότερη, ωστόσο το φαινόμενο της ολίσθησης κατάστασης είναι πιο σημαντικό. Οπότε  $0.5\text{V} < V_{\text{SET}} < 50\text{V}$ . Αυτή η έκφραση παράγει μόνο ένα κάτω όριο της  $V_{\text{SET}}$ , αφού το άνω όριο είναι σημαντικά υψηλότερο από τις πρακτικές παροχές τάσεων στα chip. Για ένα memristor ελεγχόμενο από το ρεύμα (όπως το μοντέλο TEAM), δεν είναι ρεαλιστικό να καθορίσουμε μία συγκεκριμένη τάση κατωφλίου (η οποία να εξαρτάται από την παροδική memristance της συσκευής). Ένας επαρκής υπολογισμός για μία ισοδύναμη τάση κατωφλίου είναι

$$V_{\text{ON}} = i_{\text{ON}} \cdot R_{\text{OFF}}$$

Όπου η  $V_{\text{ON}}$  είναι η τάση κατωφλίου, και η  $i_{\text{ON}}$  το ρεύμα κατωφλίου. Για ένα memristor με ρεύμα κατωφλίου της τάξης των 7  $\mu\text{A}$ , η ισοδύναμη τάση κατωφλίου είναι 0.7 volts. Η  $R_{\text{G}}$  θα είναι

$$1.5\text{k}\Omega < R_{\text{G}} < 33.3\text{k}\Omega.$$

Το ευρύτατα χρησιμοποιημένο μοντέλο memristor γραμμικής ολίσθησης ιόντων είναι ασύμβατο με τις IMPLY λογικές πύλες. Σε αυτό το μοντέλο, η memristance αλλάζει γραμμικά για οποιαδήποτε εφαρμοζόμενη τάση. Για το λόγο αυτό το φαινόμενο της ολίσθησης κατάστασης είναι σημαντικό και μη ανεκτό για τις IMPLY λογικές πύλες. Έτσι ένα διαφορετικό μοντέλο memristor με ρεύμα κατωφλίου, όπως το μοντέλο TEAM είναι πιο επιθυμητό. Το μοντέλο TEAM περιγράφει με λεπτομέρεια την φυσική συμπεριφορά των memristors. Οι επιλεγμένες παράμετροι του κυκλώματος για αυτό το παράδειγμα είναι  $R_{\text{ON}} = 1\text{ k}\Omega$ ,  $R_{\text{OFF}} = 100\text{ k}\Omega$ ,  $V_{\text{COND}} = 0.5\text{ V}$ ,  $V_{\text{SET}} = 1\text{ V}$ , και  $R_{\text{G}} = 10\text{ k}\Omega$ . Η SPICE προσομοίωση βασισμένη σε αυτές τις παραμέτρους για τη memristance του  $q$  απεικονίζεται στην εικόνα 24, όπου ο χρόνος εγγραφής (καθυστέρηση) αυτής της λογικής πύλης είναι 397.1 nsec και η ολίσθηση κατάστασης είναι 0.00069%, ισοδύναμη με περίπου 145,000 εκτελέσεις πριν γίνει η εναλλαγή. Ο χρόνος εγγραφής (καθυστέρηση) και η κατάσταση ολίσθησης για διαφέρουσες  $R_{\text{G}}$  και  $V_{\text{SET}}$  έχουν αποτυπωθεί στους πίνακες 3 και 4. Μία αύξηση της αντίστασης  $R_{\text{G}}$  ή μείωση της τάσης  $V_{\text{SET}}$  αυξάνει την καθυστέρηση της πύλης, αλλά χαμηλώνει το φαινόμενο της ολίσθησης κατάστασης( και αντίστροφα). Ο χρόνος εγγραφής (καθυστέρηση) και η ολίσθηση κατάστασης για διαφορετικές παραμέτρους του memristor έχουν αποτυπωθεί στον πίνακα 4. Μία αύξηση στην μη γραμμικότητα του memristor ( $\alpha_{\text{ON}}$ ) αυξάνει την καθυστέρηση της πύλης, αλλά χαμηλώνει το φαινόμενο της ολίσθησης κατάστασης (και αντίστροφα). Μία αύξηση στο  $k_{\text{on}}$  μειώνει την καθυστέρηση της πύλης χωρίς να αλλάζει το φαινόμενο της ολίσθησης κατάστασης.

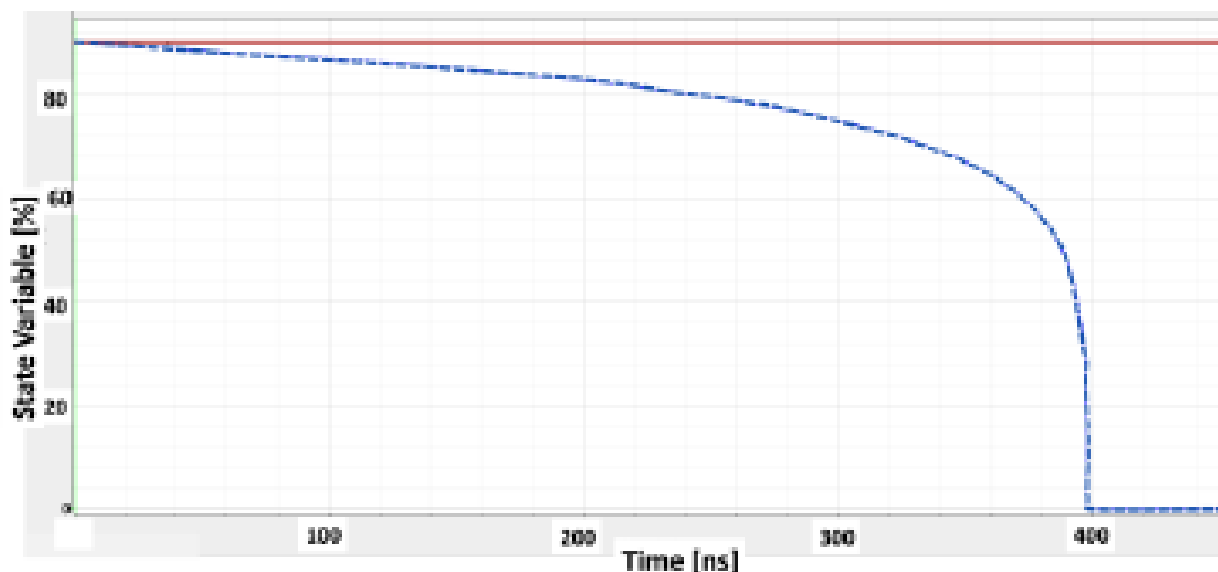
Πίνακας 3: Χρόνος εγγραφής και ολίσθηση κατάστασης για διαφορετικές τιμές της  $R_G$ . Η  $V_{COND}$  έχει καθοριστεί στα 0.5 V,  $K_{ON} = 0.05$ ,  $I_{ON} = 7 \mu A$ , και  $\alpha_{ON} = 3$ .

$R_G$ [k $\Omega$ ]	$T$ [ $\mu sec$ ]	State Drift [% $R_{OFF}$ ]	Writes Before Refresh [#]
1	0.1307	0.4655	215
3.5	0.1782	0.00244	4.09E4
5	0.2144	0.00184	5.43E4
10	0.3971	0.00069	1.45E5
15	0.7472	0.0009	1.15E6
17.5	1.038	0.00001	1.743E7
20	1.46	0	$\infty$
30	3.063	0	$\infty$

Πίνακας 4: Χρόνος εγγραφής και ολίσθηση κατάστασης για διαφορετικές τιμές της  $V_{SET}$  και των παραμέτρων του memristor.

Parameter	$T$ [ $\mu sec$ ]	State Drift [% $R_{OFF}$ ]	Writes Before Refresh [#]
Base	0.3971	0.00069	1.45E5
$V_{SET} = 1.2$ V	0.0945	0.31208	320
$k_{on} = 0.1$	0.1986	0.00069	1.45E5
$k_{on} = 0.01$	1.9866	0.0007	1.44E5
$\alpha_{on} = 1$	0.1587	0.3669	273
$\alpha_{on} = 4$	0.7927	0.0004	2.52E5





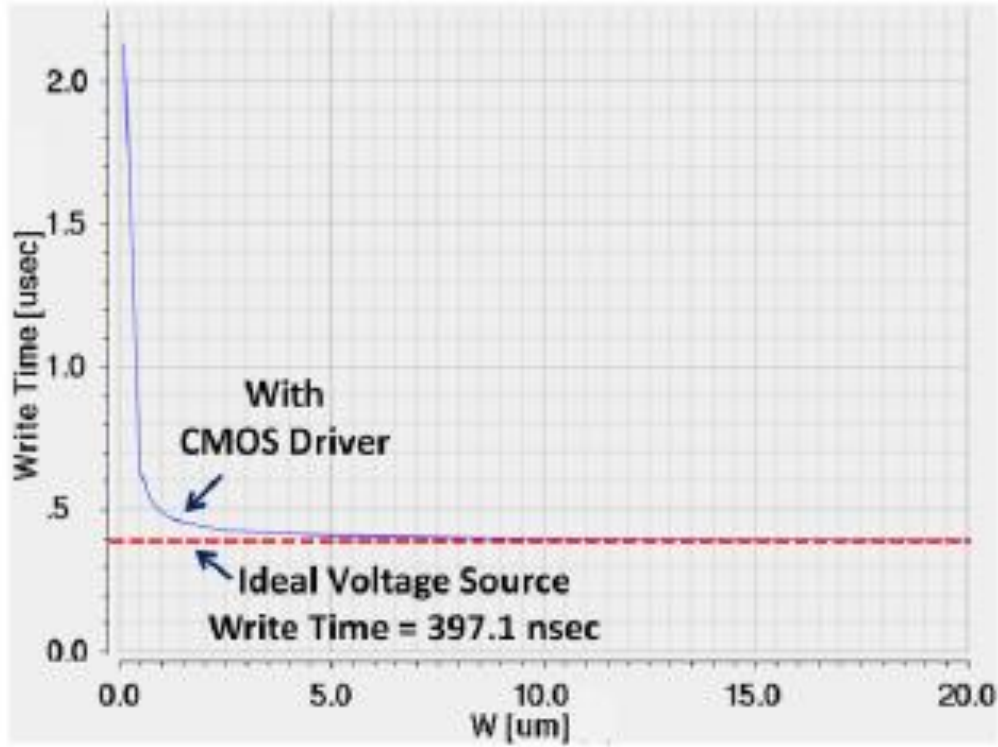
Εικόνα 24: Η μεταβλητή κατάσταση της  $q$  όταν εφαρμόζεται μια IMPLY λογική πύλη για τις περιπτώσεις 1 (διακεκομμένη γραμμή) και 3 (συνεχής γραμμή). Οι παράμετροι του κυκλώματος είναι  $V_{SET} = 1V$ ,  $V_{COND} = 0.5V$ , και  $R_G = 10k\Omega$ . Οι παράμετροι του memristor είναι  $k_{on} = 0.05$ ,  $i_{on} = 7 \mu A$ , και  $\alpha_{on} = 3$ . Η καθυστέρηση της IMPLY λογικής πύλης είναι 397.1 ns και η ολίσθηση κατάστασης είναι 0.0007%, ισοδύναμη με 145,000 εκτελέσεις πριν την ανάγκη για ανανέωση.

### 3.1.9 Παραλλαγές στην $V_{SET}$ και $V_{COND}$

Στην προηγούμενη ενότητα, έγινε η υπόθεση ότι χρησιμοποιούνται οι ιδανικές πηγές για τις  $V_{SET}$  και  $V_{COND}$ . Πρακτικές εφαρμογές, ωστόσο, υποφέρουν από εναλλαγές στο επίπεδο τάσης, κυρίως λόγω της αντίστασης των CMOS drivers. Οι CMOS drivers προσθέτουν αντίσταση στις σειρές με κυκλώματα και αλλάζουν τις εφαρμοζόμενες τάσεις. Αυτές οι πτώσεις τάσης αλλάζουν την απόδοση (όπως έχει καθοριστεί από την περίπτωση εισόδου 1) και την ολίσθηση κατάστασης (όπως αυτή έχει καθοριστεί από την περίπτωση εισόδου 3).

Για να εκτιμήσουμε την επιρροή των CMOS drivers στην απόδοση και την ολίσθηση κατάστασης, η IMPLY λογική πύλη προσομοιώνεται με παρόμοιες παραμέτρους κυκλώματος με τις προηγούμενες ενότητες. Η ισοδύναμη αντίσταση του CMOS driver για διάφορα πλάτη CMOS αποτυπώνονται στον πίνακα 5. Ο χρόνος εγγραφής για διαφορετικά πλάτη driver απεικονίζεται στην εικόνα 25. Για W/L αναλογία της τάξης του 10, ο χρόνος εγγραφής της IMPLY λογικής πύλης με DMOS drivers αυξάνεται κατά 15%, σε σύγκριση με ιδανικές πηγές τάσης. Για W/L αναλογία της τάξης του 75, η αύξηση του χρόνου εγγραφής είναι αμελητέα (μικρότερη από 1%).

Για να αξιολογήσουμε την αλλαγή στο φαινόμενο της ολίσθησης τάσης, η IMPLY λογική πύλη αξιολογείται για την περίπτωση εισόδου 3. Η διαφορά στην ολίσθηση κατάστασης αποτυπώνεται στον πίνακα 6, εμφανίζοντας αμελητέα διαφορά για όλες τις αναλογίες W/L. Για να ξεπεραστούν οι μεταβολές στην πηγή τάσης, οι εφαρμοζόμενες τάσεις ( $V_{SET}$  και  $V_{COND}$ ) μπορούν να αυξηθούν. Εναλλακτικά, η αντίσταση του κυκλώματος μπορεί να αυξηθεί, αυξάνοντας το  $R_G$ , ή χρησιμοποιώντας memristors με υψηλότερη  $R_{ON}$  και  $R_{OFF}$  ή η αντίσταση του CMOS driver μπορεί να μειωθεί αυξάνοντας την αναλογία W/L.



Εικόνα 25: Ο χρόνος εγγραφής μίας IMPLY λογικής πύλης με CMOS drivers για διάφορα CMOS πλάτη (μπλε γραμμή) σε σύγκριση με τον χρόνο εγγραφής με ιδανική πηγή τάσης (κόκκινη διακεκριμένη γραμμή).

**Πίνακας 5: Η αντίσταση ενός CMOS driver για 0.12 UM CMOS διαδικασία.**

W [ $\mu\text{m}$ ]	W/L	CMOS Driver Resistance [ $\Omega$ ]	Voltage Drop with a Load of 100 k $\Omega$
0.13	1	12.8k	11.33%
0.3	2.3	6.4k	6.00%
0.5	3.8	3.8k	3.67%
0.75	5.8	2.5k	2.42%
1	7.7	1.8k	1.83%
1.3	10	1.4k	1.33%
2.5	19.2	708	0.67%
5	38.5	349	0.33%
10	76.9	173	0.17%
20	153.8	86	0.08%

**Πίνακας 6: Ολίσθηση τάσης για IMPLY λογική πύλη με CMOS BUFFERS σε σύγκριση με ιδανικές πηγές τάσης για διάφορες αναλογίες W/L.**

W [ $\mu\text{m}$ ]	W/L	Difference in the State
0.13	1	-0.000502%
0.3	2.3	-0.000150%
0.5	3.8	0.000009%
0.75	5.8	0.000053%
1	7.7	0.000059%
1.3	10	0.000056%
2.5	19.2	0.000038%
5	38.5	0.000021%
10	76.9	0.000011%
20	153.8	0.000006%

### 3.1.10 Πλήρης αθροιστής οχτώ bit

Η IMPLY μαζί με τη FALSE (την λειτουργία η οποία βγάζει πάντα σαν αποτέλεσμα μηδέν) παρέχουν ένα ολοκληρωμένο λογικό οικοδόμημα. Αν και οποιαδήποτε Boolean λειτουργία μπορεί να εκτελεστεί, μία αποδοτική διαδικασία είναι αναγκαία για την μείωση του χώρου και του υπολογιστικού χρόνου. Σε αυτήν την ενότητα θα γίνει μια παρουσίαση ενός πλήρους αθροιστή οχτώ bit για να γίνει μια μελέτη των διάφορων σχεδιαστικών περιορισμών και θεμάτων για γενικές Boolean λειτουργίες. Σε αυτή τη μελέτη, παρουσιάζονται τρεις προσεγγίσεις: πρώτα παρουσιάζεται ένας γενικός αλγόριθμος, ο οποίος απαιτεί μια μεγάλη ακολουθία και μόνο δύο επιπλέον memristors. Δύο ακόμα προσεγγίσεις η συριακή και η παράλληλη λαμβάνονται επίσης υπόψη.

### 3.1.11 Γενικές Boolean λειτουργίες

Ένας αλγόριθμος για την εκτέλεση οποιασδήποτε Boolean λειτουργίας χρησιμοποιώντας μόνο IMPY και FALSE έχει ήδη προταθεί. Ο αλγόριθμος αυτός απαιτεί  $n + 3$  memristors για οποιαδήποτε λειτουργία  $f: B^n \rightarrow B$ . Αν και ο αλγόριθμος αυτός είναι επαρκής από άποψη χώρου (ο αριθμός των memristors για τον υπολογισμό μιας λειτουργίας), είναι μη αποδοτικός από άποψη υπολογιστικού χρόνου και απαιτεί  $O(2^{kn})$  υπολογιστικά βήματα, όπου  $n$  είναι ο αριθμός των memristors εισόδου και  $k$  είναι ο αριθμός των memristors για την υπολογιστική διαδικασία. Μια διαφορετική προσέγγιση λοιπόν απαιτείται για τη βελτίωση του υπολογιστικού χρόνου. Αυτή η νέα προσέγγιση παρουσιάζεται σε αυτή την ενότητα.

Αρκετές Boolean λειτουργίες οι οποίες εκτελούνται από τις IMPLY και FALSE αποτυπώνονται στον πίνακα 7. Αυτές οι λειτουργίες είναι τα βασικά building blocks κάθε γενικής Boolean λειτουργίας. Η διαλογή των κατάλληλων building blocks και υπολογιστικής ακολουθίας είναι μεγάλης σημασίας όταν ο στόχος είναι η ελαχιστοποίηση του αριθμού των υπολογιστικών βημάτων και των memristors. Για τη μείωση του αριθμού των υπολογιστικών βημάτων, μπορεί να χρησιμοποιηθεί ο παραλληλισμός, όπου πολλές IMPLY και FALSE λειτουργίες εκτελούνται κατά τη διάρκεια του ίδιου κύκλου ρολογιού. Αφού η λειτουργία ολοκληρώνεται μέσα στη crossbar κατασκευή, η τοπολογία ολοκλήρης της σειράς πρέπει να ληφθεί υπόψη, συμπεριλαμβανομένων και των sneak paths. Άλλες μέθοδοι για παραλληλισμό οι οποίες δεν υποφέρουν από sneak paths χρησιμοποιούν μονοπολικά memristors ή εναλλακτικά ενσωματώνουν διακόπτες ανάμεσα στις σειρές, οι οποίες αποκλίνουν από την δομή του crossbar.

Επίσης μερικές φορές είναι απαραίτητη η αντιγραφή μιας τιμής από το ένα κελί μνήμης στο άλλο. Η διαδικασία της αντιγραφής είναι επίσης απαραίτητη όταν τα δεδομένα χρησιμοποιούνται πολλές φορές, καθώς η καταστροφή της εισόδου είναι μη επιθυμητή, ή επειδή υπάρχει η ανάγκη μεταφοράς δεδομένων σε διαφορετικές σειρές μέσα στο crossbar. Η λειτουργία της αντιγραφής αποτυπώνεται επίσης στον πίνακα 7.

**Πίνακας 7: Βασικές Boolean λειτουργίες βασιζόμενες μόνο στις IMPLY και FALSE.**

<i>Structure</i>	<i>Operation</i>	<i>Comments</i>
$0 \rightarrow q$	$q' = 1$	
$1 \rightarrow q$	$q' = q$	
$p \rightarrow 0$	$q' = \text{NOT}(p)$	
$(A \rightarrow (B \rightarrow 0)) \rightarrow 0$	$q' = A \text{ AND } B$	Result in different memristor than the inputs
$(A \rightarrow 0) \rightarrow B$	$B' = A \text{ OR } B$	
$(A \rightarrow B) \rightarrow ((B \rightarrow A) \rightarrow 0)$	$q' = A \text{ XOR } B$	Requires copying of the inputs, separate output $q$
FALSE(B), FALSE(C), $A \rightarrow C$ , $C \rightarrow B$	$B' = A$	Copy operation – copy A to B

### 3.1.12 CMOS πλήρης αθροιστής

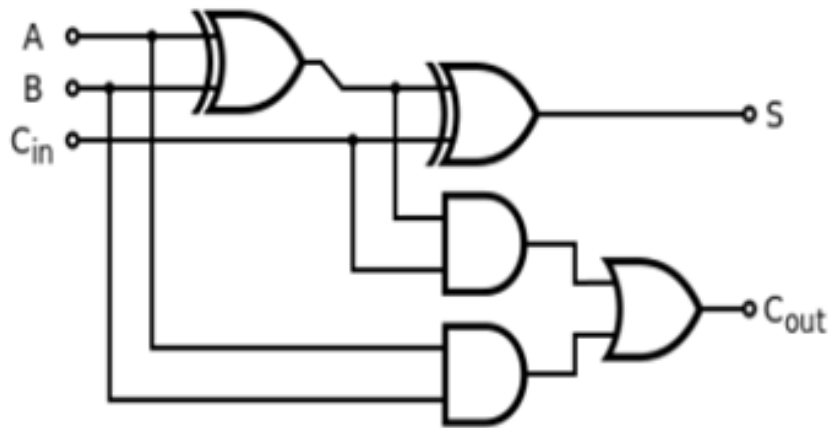
Η είσοδος του πλήρους αθροιστή είναι δύο αριθμοί των οχτώ bit και το αποτέλεσμα είναι ένας αριθμός των οχτώ bit  $S_7, S_6, \dots, S_0$  και ενός bit carry  $C_{out}$ . Η βασική δομή ενός οχτώ-bit CMOS ripple carry αθροιστή αποτελείται από οχτώ πλήρεις αθροιστές, όπου η λογική λειτουργία κάθε

$$S_i = A_i \oplus B_i \oplus C_i,$$

$$C_{out} = (A_i \cdot B_i) + (C_i \cdot (A_i \oplus B_i)).$$

αθροιστή είναι

Ένας μόνο οχτώ-bit CMOS πλήρης αθροιστής αποτελείται από 400 CMOS transistors, όπως απεικονίζεται στην εικόνα 26 για έναν βασικό πλήρη αθροιστή.



Εικόνα 26: Πλήρης αθροιστής αποτελούμενος από δύο XOR πύλες, δύο AND πύλες. Και μία OR πύλη.

### 3.1.13 IMPLY πλήρης αθροιστής

Υπάρχουν πολλές προσεγγίσεις στο σχεδιασμό ενός οχτώ-bit πλήρους αθροιστή βασισμένο μόνο σε IMPLY και FALSE λειτουργίες. Δύο από αυτές τις προσεγγίσεις είναι η συριακή και η παράλληλη. Για να αξιολογήσουμε αυτές τις προσεγγίσεις, συγκρίνονται ο συνολικός αριθμός των memristors και ο συνολικός αριθμός των υπολογιστικών βημάτων. Ο γενικός αλγόριθμος απαιτεί 712 υπολογιστικά βήματα, ενώ η συριακή προσέγγιση χαμηλώνει στον υπολογιστικό χρόνο στα 232 υπολογιστικά βήματα με περίπου τον ίδιο αριθμό memristors, και η παράλληλη προσέγγιση έχει την καλύτερη επίδοση με 58 υπολογιστικά βήματα αλλά απαιτεί τον διπλάσιο αριθμό memristors. Μία σύγκριση ανάμεσα στις προσεγγίσεις αποτυπώνεται στον πίνακα 8.

Για την εκτέλεση μίας XOR λειτουργίας, απαιτούνται δύο memristors M1 και M, όπου η πλήρης ακολουθία όπως αποτυπώνεται στον πίνακα 7, είναι

$$\begin{aligned}
 A \text{ XOR } B: & \quad \text{FALSE}(M1), \text{ FALSE}(S), A \rightarrow S, S \rightarrow M1 \\
 & \quad \text{FALSE}(M2), \text{ FALSE}(S), B \rightarrow S, S \rightarrow M2 \\
 & \quad B \rightarrow M1, \text{ FALSE}(S), M1 \rightarrow S \\
 & \quad A \rightarrow M2, M2 \rightarrow S.
 \end{aligned}$$

Οι πρώτες δύο σειρές λειτουργίες αντιγραφής του A και B, αντίστοιχα στο M1 και M2 καθώς η IMPLY λειτουργία καταστρέφει και τις δύο εισόδους. Για την εκτέλεση του  $S_i$ , η εκτελεστική διαδικασία χωρίζεται σε XOR λειτουργίες, όπου

$$S_i = (A_i \oplus B_i) \oplus C_i.$$

Αυτή η εκτέλεση απαιτεί δύο memristors και 26 υπολογιστικά βήματα για το  $S_i$ , ενώ το ενδιάμεσο XOR του  $A_i$  και  $B_i$  χρησιμοποιείται επίσης και για  $C_{out,i}$ , όπου

$$C_{out,j} = (A_i \rightarrow (B_i \rightarrow '0')) \rightarrow ((C_i \rightarrow ((A_i \oplus B_i) \rightarrow '0')) \rightarrow '0').$$

Υπάρχουν πολλές πιθανές ακολουθίες για την εκτέλεση του  $C_i$  με τη χρήση τριών memristors για να μειωθεί ο αριθμός των υπολογιστικών βημάτων.

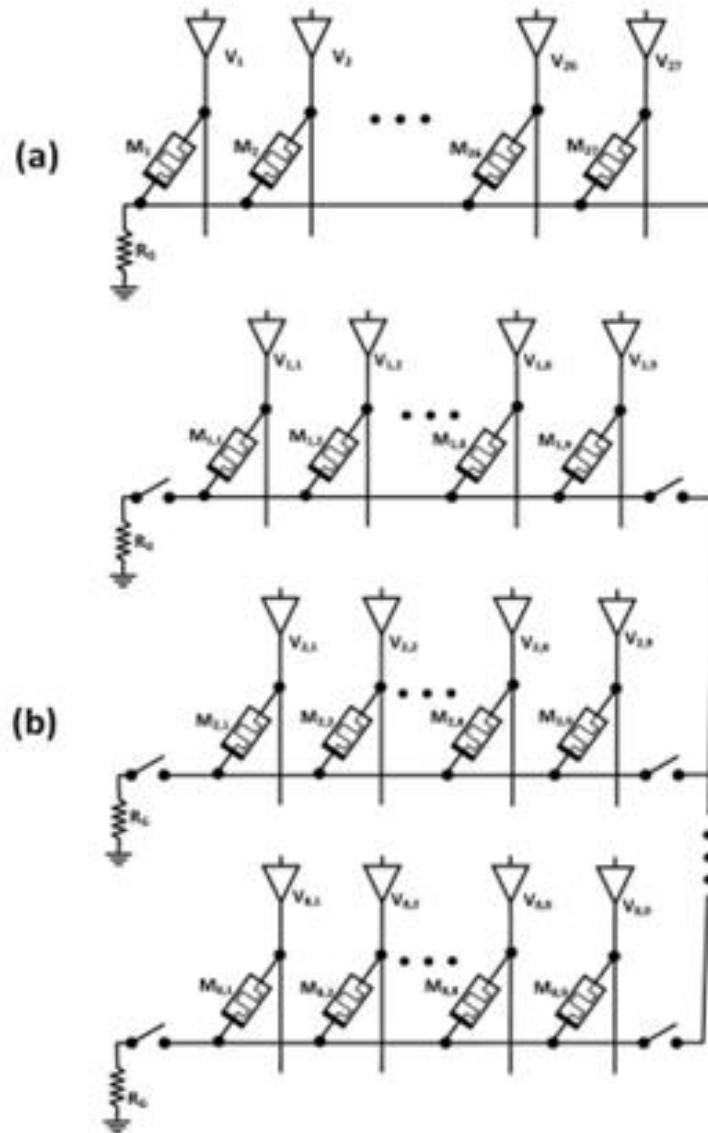
Για τον πλήρη αθροιστή οχτώ-bit, δύο προσεγγίσεις έχουν εξεταστεί σε αυτή τη μελέτη. Η συριακή προσέγγιση εκτελεί μία λειτουργία κάθε κύκλο ρολογιού - IMPLY ή FALSE. Για την συριακή προσέγγιση, όλα τα memristors είναι στην ίδια σειρά, όπως απεικονίζεται στην εικόνα 27a. Στην παράλληλη προσέγγιση, ανεξάρτητες λειτουργίες εκτελούνται κατά τη διάρκεια του ίδιου κύκλου ρολογιού, μειώνοντας τον αριθμό των απαιτούμενων υπολογιστικών σταδίων. Για την παράλληλη προσέγγιση, κάθε bit στον πλήρη αθροιστή είναι μια διαφορετική σειρά, όπως απεικονίζεται στην εικόνα 27b. Το υπόλοιπο περνάει ανάμεσα στις διάφορες σειρές και οι FALSE λειτουργίες εκτελούνται στιγμιαία για πολλά memristors. Η παράλληλη προσέγγιση απαιτεί μερικές τροποποιήσεις οι οποίες διαφέρουν από την crossbar δομή, προσθέτοντας συνδέσεις ανάμεσα στις σειρές του crossbar. Αυτές οι τροποποιήσεις επίσης εξαλείφουν το φαινόμενο sneak path ενώ αυξάνουν τον χώρο σε σύγκριση με ένα συμβατικό crossbar.

### 3.1.14 Πέρα από τον Von Neumann – λογική μέσα στη μνήμη

Η IMPLY λογική είναι φυσική μέθοδος για την εκτέλεση λογικών λειτουργιών μέσα στα memristors. Η memristor-based IMPLY λογική έχει την ίδια crossbar δομή με μία memristor-based μνήμη και για το λόγο αυτό επιτρέπει την δυνατότητα εκτέλεσης λογικών λειτουργιών μέσα στη μνήμη με τα ίδια κελιά που χρησιμοποιούνται για τη φύλαξη δεδομένων. Αυτός ο συνδυασμός επιτρέπει καινοτόμες αρχιτεκτονικές υπολογιστών, πέρα από την κλασική von Neumann αρχιτεκτονική όπου οι υπολογιστικές λειτουργίες και η αποθήκευση δεδομένων διαχωρίζεται.

Για αυτές τις νέες αρχιτεκτονικές, μέρος των υπολογισμών επιτυγχάνεται μέσα στη μνήμη, χωρίς καθόλου διαχωρισμό από τις λειτουργίες διαβάσματος και εγγραφής δεδομένων. Αυτές οι αρχιτεκτονικές είναι πιο κατάλληλες για μαζικές παράλληλες εφαρμογές, όπου τεράστιος όγκος δεδομένων πρέπει να επεξεργαστεί. Στην von Neumann αρχιτεκτονική για μαζικές παράλληλες εφαρμογές, η μεταφορά δεδομένων είναι ένα μεγάλο ζήτημα και καταναλώνει μεγάλη ενέργεια. Σε αυτές τις νέες αρχιτεκτονικές, η μνήμη και οι λογικές λειτουργίες είναι στο ίδιο crossbar, πράγμα το οποίο σημαίνει ότι δεν απαιτείται καμία μεταφορά δεδομένων, και η καθυστέρηση μαζί με την απαιτούμενη ενέργεια μειώνονται δραματικά, αν και η καθυστέρηση της IMPLY λογικής είναι μεγαλύτερη από αυτή της CMOS λογικής.

Σε αυτές τις καινοτόμες τεχνολογίες, η memristive μνήμη παίζει δύο ρόλους – ως μνήμη για την αποθήκευση δεδομένων και ως υπολογιστική μονάδα. Η λειτουργία ενός συγκεκριμένου memristor μπορεί να καθοριστεί δυναμικά. Κάθε memristor μπορεί να λειτουργήσει είτε ως ένα κελί μνήμης ή ως μέρος μιας IMPLY λογικής πύλης σε διαφορετικά στάδια μιας λειτουργίας. Το μέγεθος της μνήμης και της υπολογιστικής μονάδας είναι ευέλικτο και μπορεί να διαφέρει για διαφορετικές εφαρμογές. Μία memristor-based μνήμη απαιτεί έναν σχετικά περίπλοκο χειριστή ο οποίος θα είναι σε θέση να δρα ως ένας κανονικός χειριστής μνήμης ο οποίος θα είναι και σε θέση να στέλνει σήματα χειρισμού ( $V_{SET}$  και  $V_{COND}$ ) στις IMPLY λογικές πύλες. Αυτή η νέα αρχιτεκτονική απαιτεί ένα νέο σετ οδηγιών για τις λογικές λειτουργίες μέσα στη μνήμη.



Εικόνα 27: Ένας πλήρης αθροιστής οχτώ-bit για (a) συριακή προσέγγιση, και (b) παράλληλη προσέγγιση. Για τη συριακή προσέγγιση χρησιμοποιούνται 27 memristors στην ίδια σειρά μίας στάνταρ crossbar κατασκευής. Η παράλληλη προσέγγιση απαιτεί μια πιο περίπλοκη crossbar δομή, όπου υπάρχει μια switched σύνδεση ανάμεσα στις σειρές. Κάθε εκτέλεση bit γίνεται σε διαφορετική σειρά με τη χρήση εννέα memristors.



### 3.1.15 Συμπέρασμα για IMPLY

Μία IMPLY λογική πύλη είναι ένας φυσικός τρόπος για την εκτέλεση λογικών λειτουργιών με memristors. Αυτή η λογική πύλη μπορεί να ολοκληρωθεί με μία memristor-based μνήμη και μαζί με τη FALSE, παρέχουν μια ολοκληρωμένη λογική οικογένεια. Η memristive λογική πύλη. Αυτή η memristive λογική οικογένεια καθιστά ικανές και νέες μη-von Neumann αρχιτεκτονικές οι οποίες μπορεί να ανοίξουν μια νέα εποχή στην αρχιτεκτονική υπολογιστών.

## 3.2 MAGIC – Memristor Aided Logic

### 3.2.1 Λειτουργική αρχή της MAGIC

Η MAGIC απαιτεί μόνο memristors μέσα στις λογικές πύλες. Η λογική κατάσταση σε μία MAGIC πύλη αναπαριστάται ως μία αντίσταση, όπου οι υψηλές και οι χαμηλές αντιστάσεις θεωρούνται αντίστοιχα ως λογικά μηδέν και ένα (για απλοποίηση, η αντίσταση του λογικού μηδέν και του λογικού ένα θεωρούνται αντίστοιχα ως,  $R_{off}$  και  $R_{on}$ ). Οι είσοδοι και έξοδος των λογικών πυλών είναι οι λογικές καταστάσεις των memristors. Σε αντίθεση με μία IMPLY λογική πύλη, ξεχωριστά memristors απαιτούνται για είσοδο και έξοδο. Οι είσοδοι των MAGIC πυλών είναι οι αρχικές λογικές καταστάσεις των memristor εισόδου και το αποτέλεσμα είναι η τελική λογική κατάσταση του memristor. Αυτή η γενική ιδέα απεικονίζεται στην εικόνα 29a για μία MAGIC NOR πύλη, όπου οι αρχικές αντιστάσεις των memristors in1 και in2 είναι οι είσοδοι της πύλης και η τελική αντίσταση του memristor out είναι το αποτέλεσμα της πύλης.

Η λειτουργία μιας MAGIC πύλης αποτελείται από δύο συριακά στάδια. Το πρώτο στάδιο αρχικοποιεί το memristor εξόδου σε μία γνωστή λογική κατάσταση. Στο δεύτερο στάδιο της λειτουργίας, μια τάση  $V_0$  εφαρμόζεται στη λογική πύλη. Καθώς εφαρμόζεται η  $V_0$ , η τάση στο memristor εξόδου εξαρτάται από τη λογική κατάσταση των memristors εισόδου και εξόδου. Τα μη γραμμικά χαρακτηριστικά του memristor, κυρίως τα ρεύματα ή οι τάσεις κατωφλίου εκμεταλλεύονται στο έπακρον για να διατηρηθεί η σωστή λειτουργία. Για συγκεκριμένους συνδυασμούς εισόδου, η τάση είναι επαρκής για να αλλάξει η λογική κατάσταση του memristor εξόδου, δηλαδή η τάση/ρεύμα είναι μεγαλύτερη από το κατώφλι τάση/ρεύμα, ενώ για άλλους συνδυασμούς εισόδου, η έξοδος παραμένει στην αρχική κατάσταση, δηλαδή η τάση/ρεύμα του memristor είναι κάτω από την τάση ή το ρεύμα κατωφλίου. Αξίζει να σημειωθεί ότι σε συγκεκριμένες περιπτώσεις η πλήρης εναλλαγή δεν είναι εφικτή για memristors με ρεύμα κατωφλίου.

Η αρχικοποίηση του memristor μπορεί να επιτευχθεί με πολλούς τρόπους. Για παράδειγμα, είναι δυνατή η χρήση μίας τοπολογίας κυκλώματος για παραμετροποίησιμα αναλογικά κυκλώματα [17], όπως απεικονίζεται στην εικόνα 28. Για τις MAGIC πύλες μέσα στη μνήμη, η αρχικοποίηση επιτυγχάνεται ως μία συνήθης λειτουργία εγγραφής μέσα στα κελιά μνήμης.

### 3.2.2 Magic Nor Πύλη

Μία πύλη NOR δύο εισόδων αποτελείται από δύο memristor εισόδου (in1, in2) συνδεδεμένα παράλληλα και ένα επιπλέον memristor (out) ως την έξοδο. Ένα σχηματικό μίας NOR πύλης δύο εισόδων απεικονίζεται στην εικόνα 30α. Το αρχικό εκτελεστικό βήμα περιλαμβάνει την εγγραφή μιας χαμηλής αντίστασης στο memristor εξόδου (αρχικοποίηση στο λογικό ένα) και, αν είναι απαραίτητο την εγγραφή της τιμής εισόδου στα memristors in1 και in2. Στο τελικό εκτελεστικό βήμα, αξιολόγηση επιτυγχάνεται με την εφαρμογή ενός παλμού τάσης  $V_0$  στο gateway της λογικής πύλης.

Η εφαρμοζόμενη τάση παράγει ένα ρεύμα το οποίο διαπερνάει το κύκλωμα και εμφανίζεται στο memristor out. Για την περίπτωση κατά την οποία και τα δύο memristor εισόδου είναι λογικά μηδέν (υψηλή αντίσταση), η τάση/ρεύμα του memristor εξόδου είναι χαμηλότερη από την τάση/ρεύμα κατωφλίου του memristor. Έτσι η λογική κατάσταση του memristor εξόδου δεν αλλάζει και παραμένει στο λογικό ένα. Για όλους τους άλλους συνδυασμούς, η τάση/ρεύμα είναι μεγαλύτερη από την τάση/ρεύμα κατωφλίου του memristor. Η λογική κατάσταση του memristor εξόδου για αυτούς τους συνδυασμούς εισόδου μεταβάλλεται στο λογικό μηδέν. Η συμπεριφορά της MAGIC NOR πύλης απεικονίζεται στην εικόνα 29b. Ας υποθέσουμε ένα memristor με τάση κατωφλίου  $V_{T,OFF}$  και  $V_{T,ON}$ . Για τη σωστή συμπεριφορά του κυκλώματος, η τάση στο memristor εξόδου είναι χαμηλότερη από  $V_{T,OFF}$  όταν και οι δύο εισοδοί είναι λογικά μηδέν. Για όλους τους άλλους συνδυασμούς εισόδου, η τάση στο memristor εξόδου θα πρέπει να είναι μεγαλύτερη από  $V_{T,OFF}$ . Η ελάχιστη τάση στο memristor εξόδου να είναι μεγαλύτερη από  $V_{T,OFF}$  επιτυγχάνεται όταν μία είσοδος είναι λογικό ένα και η άλλη είσοδος είναι λογικό μηδέν. Συνδυάζοντας τις περιπτώσεις όπου η τάση στο memristor εξόδου είναι πάνω και κάτω από την τάση κατωφλίου οδηγούμαστε σε έναν σχεδιαστικό περιορισμό στην εφαρμοζόμενη τάση  $V_0$ . Ο περιορισμός είναι, υποθέτοντας

$$2V_{T,OFF} < V_0 < \frac{R_{OFF}}{2R_{ON}} \cdot V_{T,OFF}$$

$$R_{off} \gg R_{on},$$

Όταν ένα memristor εισόδου είναι λογικό μηδέν, η λειτουργία μίας MAGIC NOR μπορεί να είναι καταστροφική, αλλάζοντας την είσοδο σε λογικό ένα κατά τη διάρκεια της εκτέλεσης. Για να εξαλείψουμε το φαινόμενο της καταστροφής της εισόδου, η τάση στο memristor εισόδου πρέπει να είναι μικρότερη από την τάση κατωφλίου  $V_{T,ON}$ . Για το λόγο αυτό η μέγιστη εφαρμοζόμενη τάση για μία NOR πύλη δύο εισόδων είναι

$$V_0 < \min \left[ \frac{R_{OFF}}{2R_{ON}} \cdot V_{T,OFF}, |V_{T,ON}| \right].$$

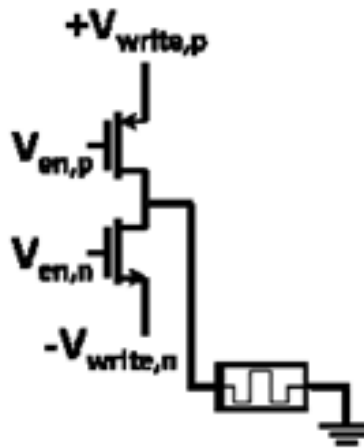
NOR λογικές πύλες πολλαπλών εισόδων (τρεις ή περισσότερες) μπορούν επίσης να παραχθούν με παρόμοιο τρόπο, όπως απεικονίζεται στην εικόνα 29c. Για  $x$  memristors εισόδου, οι σχεδιαστικοί περιορισμοί είναι

Εφαρμογές memristors σε VLSI κυκλώματα

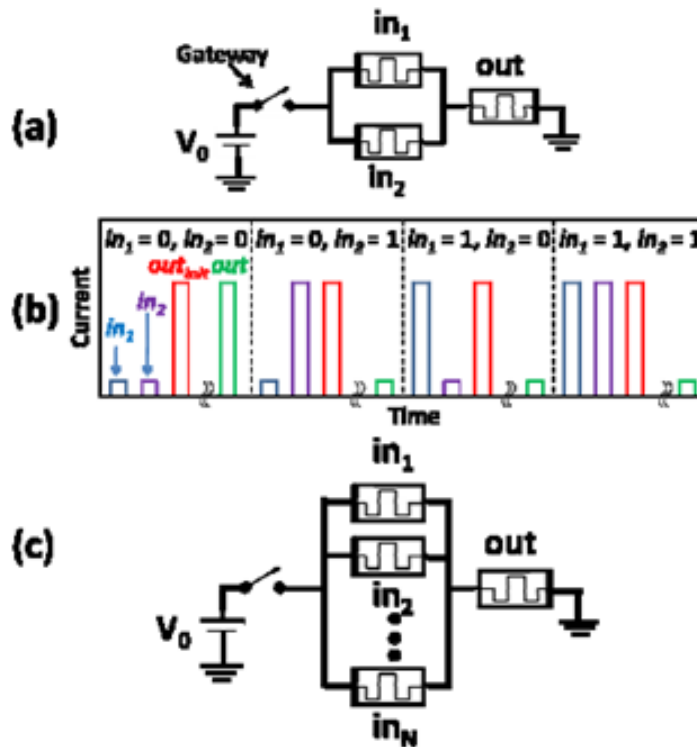
$$\frac{V_{T,OFF}}{R_{ON}} \cdot \left[ R_{ON} + \left( \frac{R_{OFF}}{\chi - 1} \right) \parallel R_{ON} \right] < V_0 < V_{T,OFF} \cdot \left[ 1 + \frac{R_{OFF}}{\chi R_{ON}} \right].$$

Για τη μη καταστρεπτική λειτουργία μίας NOR χ-εισόδων, η μέγιστη εφαρμοζόμενη τάση είναι

$$V_0 < \min \left[ V_{T,OFF} \cdot \left( 1 + \frac{R_{OFF}}{\chi R_{ON}} \right), \left( 1 + \frac{\chi R_{ON}}{R_{OFF}} \right) \cdot |V_{T,ON}| \right].$$



Εικόνα 28: Παράδειγμα τοποθέτησης ενός memristor στο στάδιο της αρχικοποίησης



**Εικόνα 29: MAGIC NOR.** (a) Σχηματικό μίας NOR πύλης δύο εισόδων. Η λογική πύλη αποτελείται από δύο memristors εισόδου  $in_1$  και  $in_2$ , και ένα memristor εξόδου  $out$ . Κατά τη διάρκεια της εκτέλεσης, μία τάση  $V_0$  εφαρμόζεται στο gateway του κυκλώματος (σημαδεμένο με ένα βέλος). (b) Προσομοιώσεις όλων των συνδυασμών για μια NOR πύλη δύο εισόδων. Τα διαφορετικά σχήματα τα ρεύματα από κάθε memristor πριν την εκτέλεση και μετά την εφαρμογή της  $V_0$ . (c) Σχηματικό μίας NOR  $N$ -εισόδων.

### 3.2.3 Magic μέσα σε Crossbar

Η RRAM συνήθως χρησιμοποιεί μία crossbar κατασκευή. Η δομή του crossbar επιτρέπει πυκνή μνήμη της τάξης των  $4 F^2$ , όπου  $F$  είναι μονάδα μέτρησης. Οι Memristive-only λογικές πύλες μέσα σε ένα crossbar μειώνουν την κατανάλωση ενέργειας και παρέχουν την ευκαιρία για νέες μη-von Neumann αρχιτεκτονικές, όπου οι λογικές λειτουργίες εκτελούνται μέσα στη μνήμη [18]. Όταν εκτελούνται λογικές λειτουργίες μέσα στη μνήμη, η είσοδος είναι τα υποθηκευμένα δεδομένα μέσα στα memristors και η έξοδος είναι τα αποθηκευμένα δεδομένα μετά την εκτέλεση. Η αρχικοποίηση της εισόδου και της εξόδου επιτυγχάνεται σαν μία συνήθη λειτουργία εγγραφής μνήμης, και η ανάγνωση του αποτελέσματος επιτυγχάνεται ως μια συνήθη λειτουργία διαβάσματος μνήμης.

Για να ενσωματώσουμε μια memristive-only λογική πύλη μέσα σε ένα crossbar, δύο απαιτήσεις πρέπει να ικανοποιηθούν: η κατασκευή και οι συνδέσεις της λογικής πύλης θα πρέπει να τοποθετηθούν μέσα σε ένα crossbar και η λογική κατάσταση της λογικής πύλης να αναπαριστάται ως μία αντίσταση, όπως στην memristive μνήμη. Μία MAGIC NOR μνήμη εκπληρώνει και τις δύο αυτές απαιτήσεις.

Αν και άλλες memristive λογικές πύλες μπορούν επίσης να τοποθετηθούν σε ένα memristive crossbar, ωστόσο, αυτές οι λογικές πύλες απαιτούν έναν επιπλέον αντιστάτη σε κάθε σειρά του crossbar. Επιπλέον, σε αντίθεση με τη NOR Boolean λειτουργία, η IMPLY λειτουργία δεν είναι λογικά ολοκληρωμένη και απαιτεί τη λειτουργία FALSE (εγγραφή ενός λογικού μηδέν σε Βασίλειος Κ. Αθανασόπουλος

ένα memristor). Μία σύγκριση ανάμεσα στην memristive IMPLY και στη MAGIC αποτυπώνεται στον πίνακα 8.

**Πίνακας 8: Σύγκριση ανάμεσα στην IMPLY και τη MAGIC**

	<b>IMPLY [11]</b>	<b>MAGIC</b>
No. of voltages	2 ( $V_{SET}$ , $V_{COND}$ )	1 ( $V_0$ )
Separate input and output	No	Yes
Basic functions	IMPLY (+ FALSE)	OR, AND, NOR, NAND, NOT
No. of memristors for NOR/NAND	3 (+ a resistor)	3
No. of clock cycles for NOR/NAND	4	2
Within memory	Yes	Yes (for NOR)
Logically complete	Requires FALSE	Yes (NOR, NAND)

### 3.2.4 Αξιολόγηση και σχεδιαστικές εκτιμήσεις για μία MAGIC NOR πύλη.

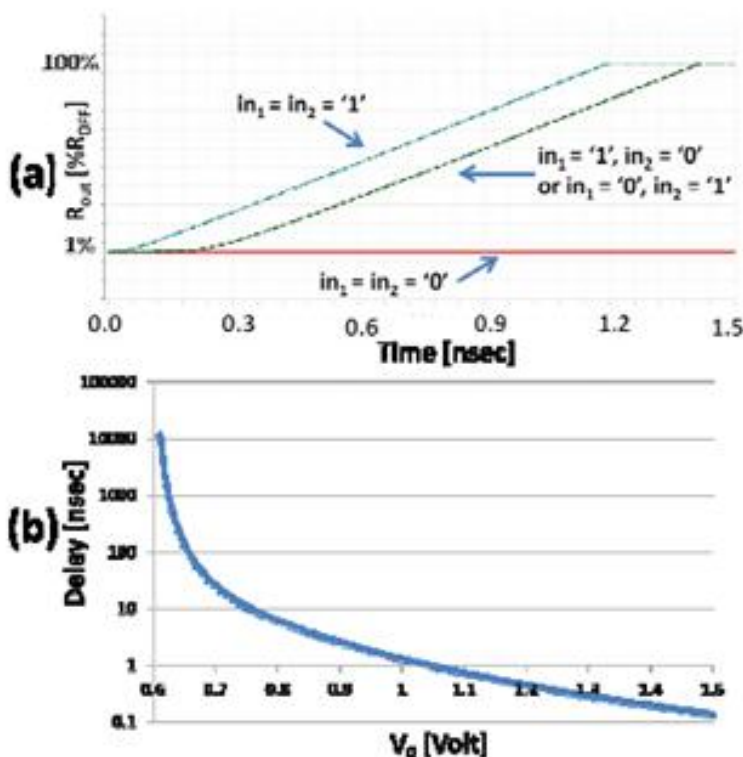
Η ταχύτητα στο SPICE για μία 0.18  $\mu\text{m}$  διαδικασία. Ένα μοντέλο memristor, το VTEAM μοντέλο [19], το οποίο επεκτείνει το TEAM μοντέλο με μία τάση κατωφλίου, χρησιμοποιείται με μία Biolek window λειτουργία. Οι παράμετροι του memristor επιλέγονται ώστε να παράγουν ένα χρόνο εναλλαγής 1 ns για ένα παλμό τάσης της τάξης του 1 volt για RESET και 2 volts για SET, και για να είναι διαθέσιμο για πρακτικές συσκευές [20]. Οι παράμετροι των προσομοιώσεων του κυκλώματος αποτυπώνονται στον πίνακα 9.

Η συμπεριφορά και η ταχύτητα μίας MAGIC NOR πύλης για διαφορετικές τιμές της  $V_0$  απεικονίζονται στην εικόνα 30. Για την αξιολόγηση της καθυστέρησης της λογικής πύλης, θεωρούμε την περίπτωση της πιο αργής εισόδου. Η καθυστέρηση μίας MAGIC NOR πύλης καθορίζεται από συνδυασμούς εισόδου  $\{1,0\}$  ή  $\{0,1\}$ .

Όπως φαίνεται στην εικόνα 30b αυξάνοντας την εφαρμοζόμενη τάση  $V_0$  μειώνεται η καθυστέρηση της λογικής πύλης. Για  $V_0$  στο 1 volt, η καθυστέρηση της λογικής πύλης είναι 1.3 ns, μια αύξηση 30% σε σύγκριση με το χρόνο εναλλαγής ενός μόνο memristor.

Πίνακας 9: Παράμετροι memristor (για το μοντέλο VTEAM)

$k_{on}$	-216.2 m/sec
$k_{off}$	0.091 m/sec
$V_{T,ON}$	-1.5 volt
$V_{T,OFF}$	0.3 volt
$x_{on}$	0
$x_{off}$	3 nm
$\alpha_{on}$	4
$\alpha_{off}$	4
$R_{ON}$	1 k $\Omega$
$R_{OFF}$	300 k $\Omega$



Εικόνα 30: Προσομοιώσεις SPICE για MAGIC NOR πύλη δύο εισόδων. (a) Memristor εξόδου για διαφορετικούς συνδυασμούς εισόδου,  $V_0 = 1$  volt. Η καθυστέρηση αξιολογείται ως ο χρόνος της εναλλαγής του memristor εξόδου σε λογικό μηδέν όταν η μία είσοδος είναι λογικό ένα και η άλλη είσοδος λογικό μηδέν και (b) καθυστέρηση για διαφορετικές τιμές της  $V_0$ .

### 3.2.5 Επιπρόσθετες MAGIC Πύλες

Με τις ίδιες σχεδιαστικές αρχές που περιγράψαμε και παραπάνω, επιπρόσθετες Boolean λειτουργίες μπορούν να προστεθούν στην MAGIC οικογένεια. Οι επιπρόσθετες MAGIC πύλες που θα περιγράψουμε εδώ δεν τοποθετούνται μέσα σε ένα crossbar (εκτός από τη NOT πύλη), αλλά μπορούν να χρησιμοποιηθούν ως μεμονωμένη λογική.

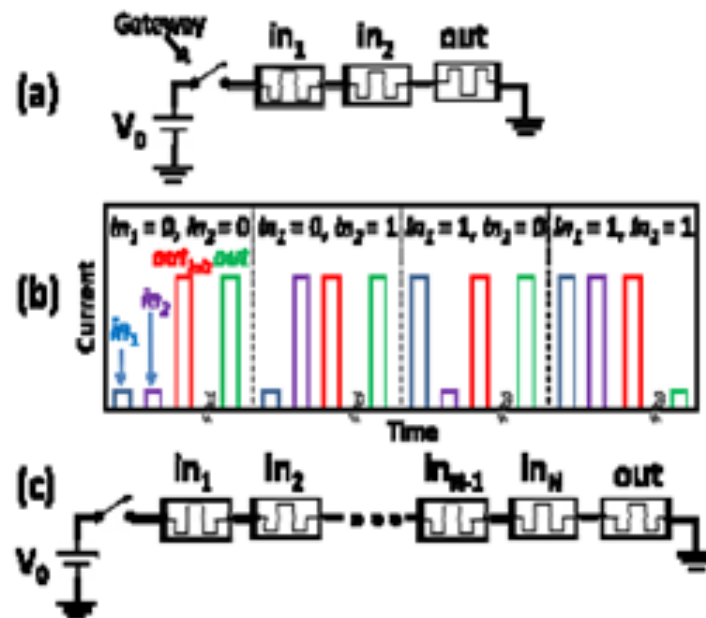
Συνδέοντας τα memristors εισόδου σε σειρά με την ίδια τοπολογία με τη MAGIC NOR πύλη παράγεται μία NAND πύλη, όπως απεικονίζεται στην εικόνα 31. Οι OR και AND λογικές πύλες έχουν παρόμοια δομή με αντίστοιχα τις NOR και NAND, εκτός βέβαια από τη πολικότητα του memristor εξόδου out. Σε αντίθεση με τις NAND και NOR, η out αρχικοποιείται σε λογικό μηδέν πριν την εκτέλεση. Το σχηματικό και η συμπεριφορά των OR και AND MAGIC πυλών απεικονίζονται στην εικόνα 32. Παρόμοια με τις MAGIC NOR και NAND πύλες, λογικές πύλες πολλών εισόδων είναι επίσης εφικτές και για MAGIC OR και AND πύλες.

Μία MAGIC NOT πύλη (αντιστροφέας) αποτελείται από ένα memristor εισόδου in και ένα memristor εξόδου out. Τα memristors είναι συνδεδεμένα σε σειρά με αντίστροφη πολικότητα σε μια συμπληρωματική κατασκευή memristor [21], όπως απεικονίζεται στην εικόνα 33a. Στο πρώτο στάδιο της εκτέλεσης, το memristor εξόδου αρχικοποιείται σε λογικό ένα. Όταν εφαρμόζεται η  $V_0$  στο gateway του κυκλώματος, ο διαιρέτης τάσης ανάμεσα στο in και στο out καθορίζει εάν η αντίσταση του memristor εξόδου αλλάζει. Για την περίπτωση κατά την οποία το in είναι λογικό μηδέν, η τάση στο out είναι κάτω από την τάση κατωφλίου και η λογική κατάσταση του out παραμένει λογικό ένα, όπως και είναι επιθυμητό. Αξίζει να σημειωθεί ότι σε αυτή την περίπτωση, η τάση στο in είναι σχετικά υψηλή και η λογική κατάσταση στο in μπορεί για αυτό το λόγο να αλλάξει σε λογικό μηδέν. Έτσι, η MAGIC NOT λειτουργία μπορεί να γίνει καταστροφική για την είσοδο εκτός αν η εφαρμοζόμενη τάση είναι κάτω από  $V_{T,ON}$ . Για την περίπτωση όπου το in είναι λογικό ένα, η τάση στο memristor out είναι αρκετή για να αλλάξει τη λογική κατάσταση του out σε λογικό μηδέν. Τα αποτελέσματα της προσομοίωσης για μια πύλη NOT απεικονίζονται στην εικόνα 34b. Μία περίληψη πολλών πυλών MAGIC, συμπεριλαμβανομένων και των σχεδιαστικών τους περιορισμών αποτυπώνονται στον πίνακα 10. Για γραμμικά memristors και memristors χωρίς τάση κατωφλίου, παρόμοια σχεδιαστικά θέματα όπως αυτά που αναφέρθηκαν και για την IMPLY λογική πρέπει να ληφθούν υπόψη, κάνοντας πιο περίπλοκη τη σχεδιαστική λειτουργία.

### 3.2.6 Συμπεράσματα MAGIC λογικής

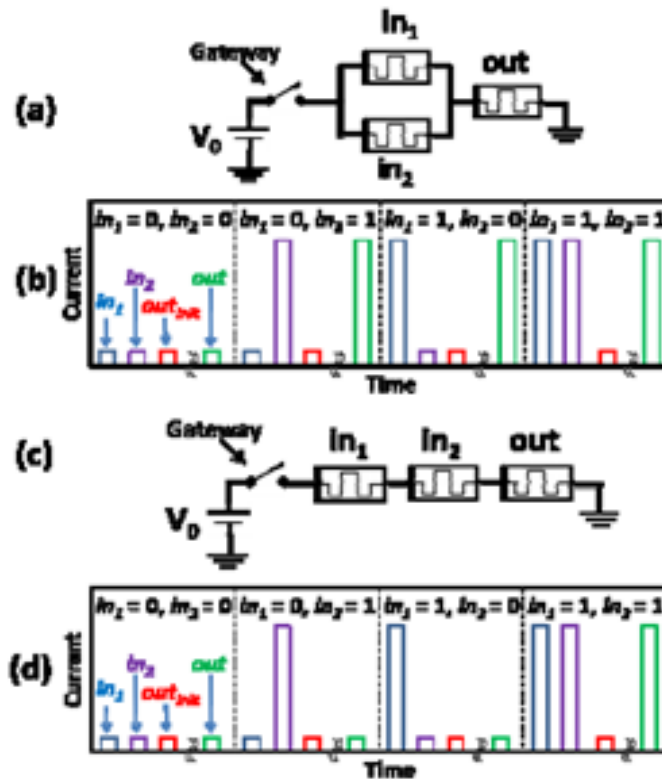
Η MAGIC είναι μια νέα μέθοδος για memristor-based λογική. Οι πέντε βασικές λογικές λειτουργίες, NOT, AND, NAND, NOR, και OR, χρησιμοποιούν απλές συνδέσεις ανάμεσα στα memristors, όπου ο αριθμός των memristors είναι ισοδύναμος με τον αριθμό των εισόδων συν ένα επιπλέον memristor στην έξοδο.

Μόνο μία εφαρμοζόμενη τάση ελέγχει αυτές τις λογικές πύλες, σε αντίθεση με τις άλλες memristor-based λογικές. Σε αντίθεση με την IMPLY λογική, η είσοδος και η έξοδος στη MAGIC είναι ξεχωριστές και η έξοδος γράφεται σε ένα ξεχωριστό αφιερωμένο σε αυτό το σκοπό memristor. Η χρήση των MAGIC NOR πυλών μέσα σε ένα memristive crossbar μπορεί να οδηγήσει σε πιο αποδοτικά συστήματα από άποψη κατανάλωσης ενέργειας, και σε νέες μη-von Neumann αρχιτεκτονικές.



Εικόνα 31: MAGIC NAND πύλη. (a) Σχηματικό μίας NAND πύλης δύο εισόδων. Η λογική πύλη αποτελείται από δύο memristor εισόδου  $in_1$  και  $in_2$ , και ένα memristor εξόδου  $out$ . Κατά την εκτέλεση, μία τάση  $V_0$  εφαρμόζεται στο gateway του κυκλώματος. (b) Τα αποτελέσματα προσομοίωσης για μία NAND πύλη δύο εισόδων για όλους τους συνδυασμούς εισόδων. (c) Σχηματικό μίας NAND πύλης  $N$ -εισόδων.

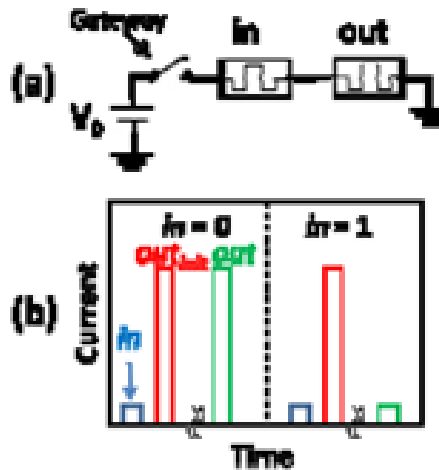




Εικόνα 32: MAGIC OR και AND πύλες. Η δομή των λογικών πυλών είναι παρόμοια με αυτή των MAGIC NOR και NAND πυλών. Το memristor εξόδου out είναι συνδεδεμένο με την ίδια πολικότητα με τα memristor εισόδου, και είναι αρχικοποιημένο σε λογικό μηδέν. (a) Σχηματικό μίας OR πύλης δύο εισόδων. (b) Αποτελέσματα προσομοίωσης για μία OR πύλη δύο εισόδων για όλους τους συνδυασμούς εισόδων. (c) Σχηματικό μίας AND πύλης δύο εισόδων. (d) Αποτελέσματα προσομοίωσης για μία AND πύλη δύο εισόδων για όλους τους συνδυασμούς εισόδων.

Πίνακας 10: Περίληψη των πυλών MAGIC

Function	Design Constraints	Design Constraints – Multiple Inputs	Within Crossbar?
NOR	$2V_{T,off} < V_0 < \min \left[ \frac{R_{off}}{2R_{on}} \cdot V_{T,off},  V_{T,on}  \right]$	$\frac{V_{T,off}}{R_{on}} \cdot \left[ R_{on} + \left( \frac{R_{off}}{Z-1} \right) \parallel R_{on} \right] < V_0 < \min \left[ V_{T,off} \cdot \left( 1 + \frac{R_{off}}{Z R_{on}} \right), \left( 1 + \frac{Z R_{on}}{R_{off}} \right) \cdot  V_{T,on}  \right]$	Yes
NAND	$3V_{T,off} < V_0 < \min \left[  V_{T,on}  \cdot \left( 2 + \frac{R_{off}}{R_{on}} \right), V_{T,off} \right]$	$(Z+1) \cdot V_{T,off} < V_0 < \min \left[  V_{T,on}  \cdot \left( 1 + \frac{Z R_{on}}{R_{off}} \right), \left( Z + \frac{R_{off}}{R_{on}} \right) \cdot V_{T,off} \right]$	No
OR	$V_{T,on} < V_0 < 1.5V_{T,on}$	$V_{T,on} < V_0 < \left( 1 + \frac{1}{Z} \right) V_{T,on}$	No
AND	$V_{T,on} < V_0 < 2V_{T,on}$	$\left( 1 + \frac{R_{on}}{R_{off}} Z \right) V_{T,on} < V_0 < \left( 2 + \frac{R_{on}}{R_{off}} (Z-1) \right) V_{T,on}$	No
NOT	$2V_{T,off} < V_0 < \frac{R_{off}}{R_{on}} \cdot \min \left( V_{T,off},  V_{T,on}  \right)$	-	Yes



Εικόνα 33: MAGIC NOT πύλη. (a) Σχηματικό μίας NOT πύλης. Η λογική πύλη αποτελείται από ένα memristor εισόδου in και ένα memristor εξόδου out, και (b) αποτελέσματα προσομοίωσης για μία NOT πύλη.

### 3.3 Υβριδική memristor-CMOS λογική (MeMOS)

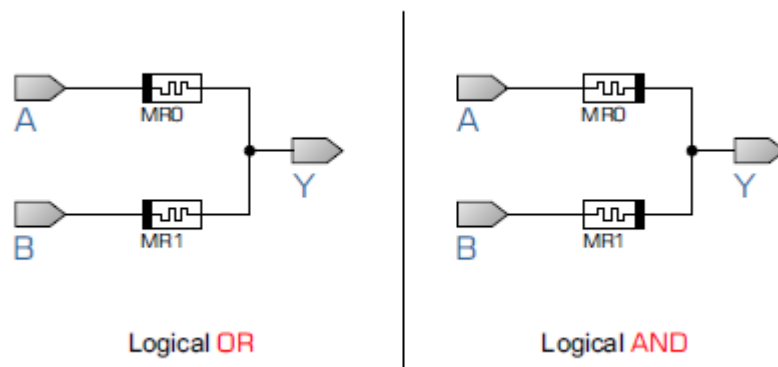
Όπως ειπώθηκε και παραπάνω η έναρξη εργασιών για την εξέλιξη και εφαρμογή των memristors έλαβε χώρα μόλις το 2008 από τα εργαστήρια της HP τα οποία παρουσίασαν το δικό τους τσιπ το οποίο αποτελούνταν από διοξείδιο του τιτανίου. Σήμερα, μετά από λίγα χρόνια έρευνας τα memristors θεωρούνται η καλύτερη εναλλακτική της σημερινής γενιάς CMOS τεχνολογίας. Πέρα από τις εφαρμογές τους για την δημιουργία μνημών ένα μεγάλο πεδίο ενδιαφέροντος είναι και αυτό του λογικού υπολογισμού με τη χρήση memristors. Οι ερευνητές έχουν προτείνει ένα σύνολο διαφορετικών μεθόδων λογικού υπολογισμού. Μία από τις πρωταρχικές μεθόδους η οποία θεωρείται και από τις σημαντικότερες είναι ο υλικός συμπερασμός με τη χρήση memristors. Η λογική υλικού συμπερασμού δείχνει να έχει ενθαρρυντικά αποτελέσματα, ωστόσο χρειάζεται ακόμα αρκετά υπολογιστικά βήματα στην λογική εκτέλεσης. Ο μεγαλύτερος περιορισμός του λογικού συμπερασμού είναι ο σχεδιασμός κυκλωμάτων διαβάσματος/γραψίματος καθώς η λογική είναι τελείως διαφορετική από αυτήν της Boolean λογικής. Επιπλέον, δεν είναι συμβατή με τη τρέχουσα γενιά της CMOS τεχνολογίας.

Η υβριδική λογική CMOS-Memristor είναι ένα υβρίδιο το οποίο αποτελείται από κοινού memristors αλλά και CMOS. Χρησιμοποιώντας αυτό το σύστημα εφαρμογής μπορούμε να εκτελέσουμε λογικές πράξεις και τα αποτελέσματα αυτών μπορούν να αποτυπωθούν σε επίπεδα τάσης. Ο όρος της νέας αυτής μεθόδου ονομάστηκε MeMOS λόγω της υβριδικής της φύσης. Μέσω αυτής της προσέγγισης οι λογικές AND και OR μπορούν να μπορούν να υπολογιστούν με τη χρήση μόνο memristors, και ο CMOS αντιστροφείας χρησιμοποιείται για την λειτουργία NOT καθώς η λειτουργία NOT δεν είναι εφικτή με τη χρήση μόνο memristors.

### 3.3.1 Λογική MeMOS

Από το μοντέλο του memristor, είναι εμφανές ότι τα memristors επιδεικνύουν μεταβλητή αντίσταση όταν το ρεύμα περνά μέσα από τη συσκευή. Οι βασικές Boolean λογικές λειτουργίες AND και OR μπορούν να αναλυθούν χρησιμοποιώντας memristors. Αν και πολλοί ερευνητές έχουν δηλώσει την λογική υλικού συμπερασμού μέσω της χρήσης memristors, ωστόσο αυτό δεν σύμφωνο με την διεργασία της τρέχουσα γενιάς CMOS. Ο υλικός συμπερασμός δουλεύει στη μεταβλητή κατάσταση, οι εισαγωγές και τα αποτελέσματα είναι οι καταστάσεις του memristor σε αντίθεση με τις τάσεις οι οποίες απαιτούνται για την διάδοση σημάτων στην CMOS διεργασία. Έτσι για την ενοποίηση του memristor με CMOS και για να λειτουργούν με τα ίδια επίπεδα τάσης, υπάρχει η ανάγκη μια υβριδικής Memristor-CMOS λογικής.

Σε αυτή τη λογική, οι τάσεις χρησιμοποιούνται ως λογική κατάσταση. Όσο για τα χαρακτηριστικά τάσης-έντασης του memristor, η βασική ιδέα της χρήσης memristors για λογικούς υπολογισμούς είναι η ιδιότητα του να μεταβάλλει την αντίσταση βάσει της διεύθυνσης της ροής του ρεύματος. Με τη χρήση αυτής της ιδιότητας είναι δυνατή η δημιουργία ενός κυκλώματος το οποίο θα λειτουργεί ως διαιρέτης τάσης [22]. Η εικόνα 34 απεικονίζει ένα κυκλώματα AND και ένα OR σχεδιασμένα με τη χρήση memristors. Το μόνο που πρέπει να κάνουμε είναι να αλλάξουμε την πόλωση των memristor για να πάρουμε τη σωστή λογική τιμή.



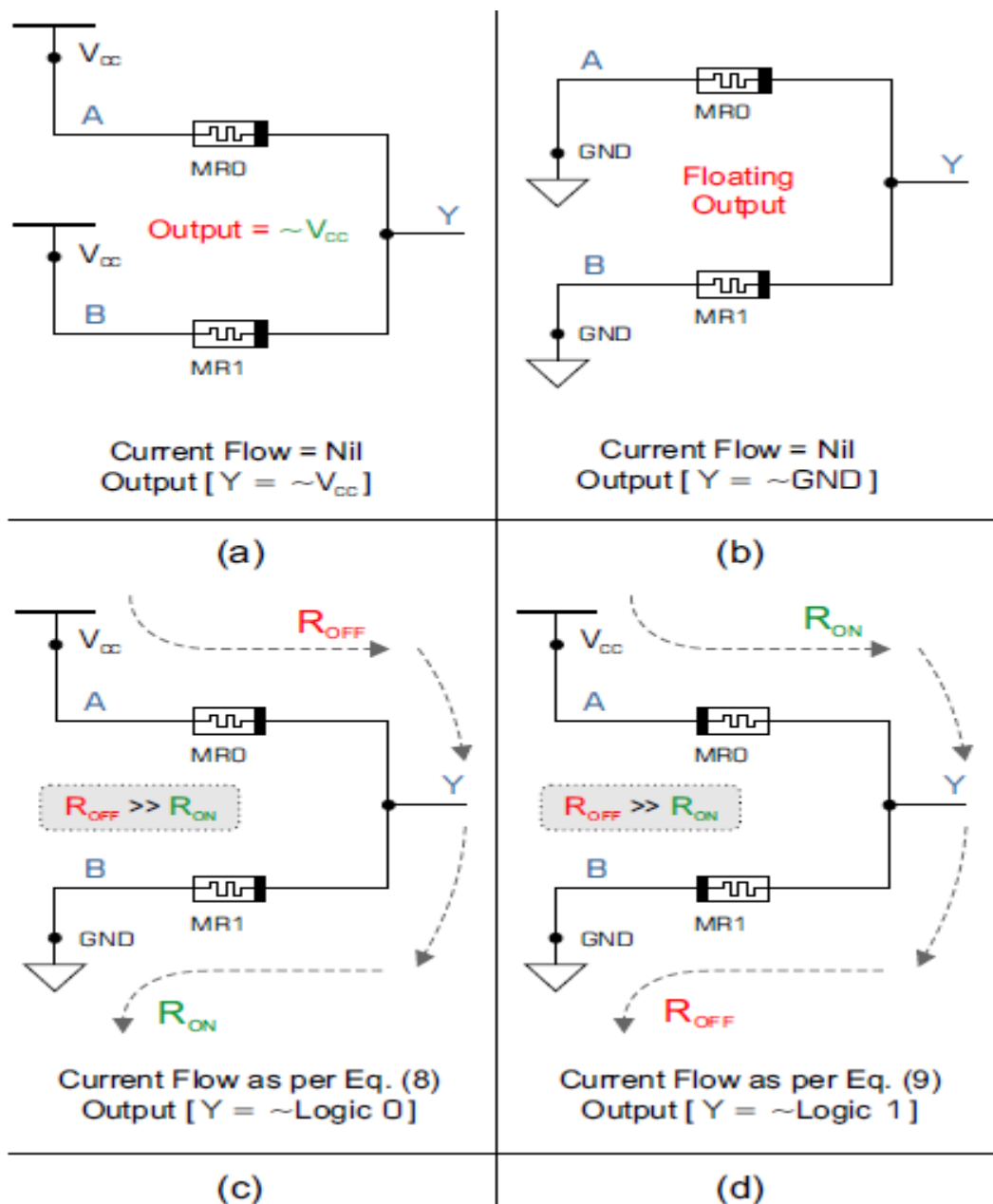
Εικόνα 34: Λογικά OR και AND κυκλώματα με τη χρήση memristor ως διαιρέτης τάσης

Η εικόνα 34 περιγράφει τον υπολογισμό την λειτουργίας AND για όλες τις περιπτώσεις μιας AND πύλης 2 εισόδων με τη χρήση μόνο memristors. Για την περίπτωση όπου το  $A=1$  και το  $B=1$ , και οι δύο εισοδοί είναι συνδεδεμένες με το  $V_{cc}$ . Όπως περιγράφεται στην εικόνα 35(a), δεν υπάρχει ροή ρεύματος μέσω του κυκλώματος και το αποτέλεσμα σε αυτή τη περίπτωση είναι  $logic = V_{cc}$  ή 1. Για την περίπτωση όπου το  $A=0$  και το  $B=0$  όπως αποτυπώνεται στην εικόνα 35(c) μπορεί να θεωρηθεί ότι οι εισοδοί είναι στο λογικό 0, και το αποτέλεσμα θα πρέπει επίσης να είναι το λογικό 0. Ξανά δεν υπάρχει ροή ρεύματος διαμέσου του

Εφαρμογές memristors σε VLSI κυκλώματα

κυκλώματος . Οι δύο αυτές περιπτώσεις είναι παρόμοιες και για το λογικό OR επίσης. Ακόμα και με την αντίθετη πόλωση των memristors, το αποτέλεσμα παραμένει το ίδιο.

Για την περίπτωση κατά την οποία οποιοδήποτε από τις δύο εισόδους είναι στο λογικό 1 και η άλλη στο λογικό 0 όπως απεικονίζεται στην εικόνα 35(b) η ροή του ρεύματος έχει την κατεύθυνση από το  $V_{CC}$  στο GND. Όταν το ρεύμα περάσει από το memristor MR0, η αντίσταση του memristor αυξάνεται σε  $R_{OFF}$ , η αντίσταση του memristor MR1 μειώνεται σε  $R_{ON}$ , και το ρεύμα διαφεύγει μέσω του κόμβου GND. Η αντίσταση των memristors είναι  $R_{OFF} \gg R_{ON}$ . Με αυτόν τον τρόπο παίρνουμε δύο αντιστάτες  $R_{OFF}$  και  $R_{ON}$  με διαφορετικές τιμές. Όσο για τον κανόνα του διαιρέτη τάσης, παίρνουμε αποτέλεσμα  $Y=0$  το οποίο ολοκληρώνει τη λογική για AND πύλες.



Εικόνα 35: Λογικός υπολογισμός με τη χρήση memristors

Ο υπολογισμός της τάσης εξόδου στο Y για τον διαιρέτη τάσης μπορεί να καθοριστεί ως

$$Y = V_{cc} \times \frac{R_{ON}}{R_{ON} + R_{OFF}}$$

Η αντίσταση  $R_{OFF}$  είναι σημαντικά μεγαλύτερη από την  $R_{ON}$ , οπότε μπορούμε να απλοποιήσουμε την εξίσωση ως

$$Y = V_{CC} \times \frac{R_{ON}}{R_{OFF}} \ll V_{CC} \approx GND$$

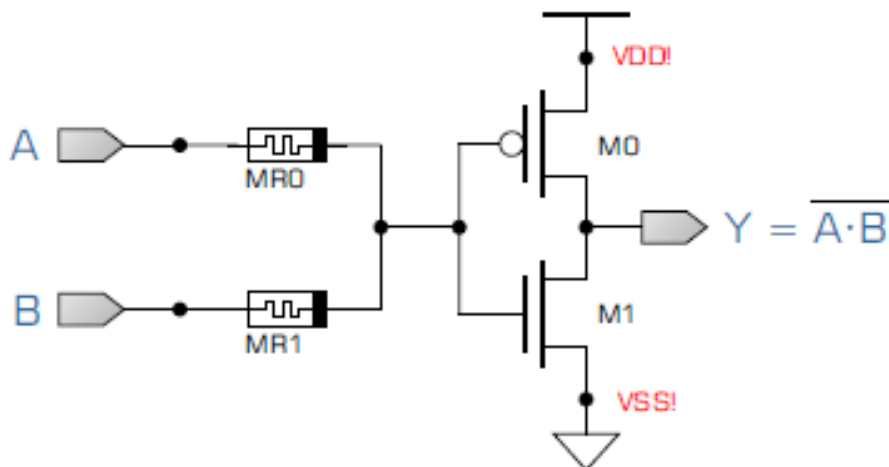
Όταν η πόλωση των memristors MR0 και MR1 αντιστραφούν, το κύκλωμα συμπεριφέρεται ως μία πύλη OR. Για εισόδους A=0, B=0 και A=1, B=1 η έξοδος Y παίρνει 0 και 1 καθώς κανένα ρεύμα δεν ρέει διαμέσου του κυκλώματος και η συμπεριφορά του παραμένει ίδια με την περίπτωση της AND πύλης. Η εικόνα 36(d) δείχνει την περίπτωση όπου οποιαδήποτε από τις δύο εισόδους είναι στο λογικό 1 και η άλλη είναι στο λογικό 0. Το ρεύμα ρέει από το  $V_{CC}$  προς το memristor MR0. Όσο το memristor είναι σε ανάστροφη πόλωση, η αντίσταση του memristor μειώνεται σε  $R_{ON}$  και με αυτόν τον τρόπο η τάση έχει αποτέλεσμα στον κόμβο εξόδου  $Y = V_{CC}$ . Το αποτέλεσμα γίνεται λογικό 1 όταν οποιαδήποτε από τις εισόδους είναι στο λογικό 1. Η αντίσταση του άλλου memristor MR1 αυξάνεται σε  $R_{OFF}$ . το αποτέλεσμα για την πύλη OR μπορεί να υπολογιστεί χρησιμοποιώντας τον κανόνα του διαιρέτη τάσης ως

$$Y = V_{CC} \times \frac{R_{OFF}}{R_{ON} + R_{OFF}} \approx V_{CC}$$

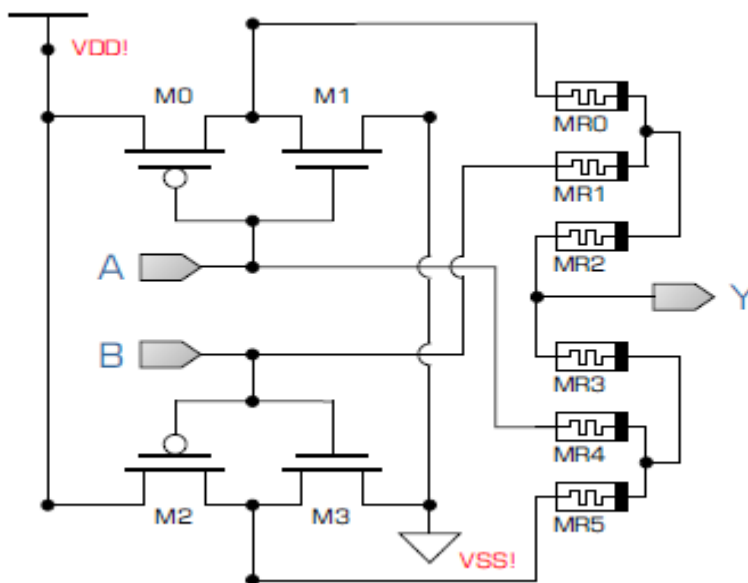
Με αυτήν την τοπολογία ακόμα και κυκλώματα με 'ν' τον αριθμό πυλών εισόδου μπορούν να υλοποιηθούν με τη χρήση memristors. Αλλά υπάρχει ακόμη το πρωταρχικό πρόβλημα το οποίο είναι η μη ολοκληρωμένη λογική οικογένεια. Χωρίς την λειτουργία NOT δεν είναι εφικτό να εκτελέσουμε Boolean λειτουργίες. Ένας CMOS αντιστροφείας μπορεί να χρησιμοποιηθεί για να εκτελέσουμε τη λειτουργία NOT. Ο CMOS αντιστροφείας έχει σχεδιαστεί χρησιμοποιώντας τεχνολογία επεξεργασίας των 180 nm. Η τάση λειτουργίας για τον CMOS αντιστροφεία είναι 1.8V. Η στατική κατανάλωση ισχύος, η καθυστέρηση καθώς και η εξοικονόμηση χώρου είναι μερικά από τα πλεονεκτήματα τα οποία προσφέρει η από κοινού CMOS και memristors. Η στρώση των memristors μπορεί να κατασκευαστεί στην κορυφή του στρώματος του CMOS.

### 3.3.2 Σχεδιασμός λογικών πυλών με χρήση της MeMOS λογικής

Όπως αναφέρθηκε και πιο πάνω η λειτουργία NOT δεν είναι δυνατόν να επιτευχθεί μόνο με τη χρήση μόνο memristors. Έτσι για να ολοκληρώσουμε την λογική οικογένεια θα πρέπει να προσθέσουμε και έναν CMOS αντιστροφεία στην έξοδο της AND πύλης έτσι ώστε να πάρουμε την λειτουργία NAND και παρομοίως την NOR λειτουργία στην περίπτωση που έχουμε OR πύλη εξόδου. Με τον ίδιο τρόπο μπορεί να δημιουργηθεί και μία XOR πύλη χρησιμοποιώντας παρόμοιες μεθόδους.



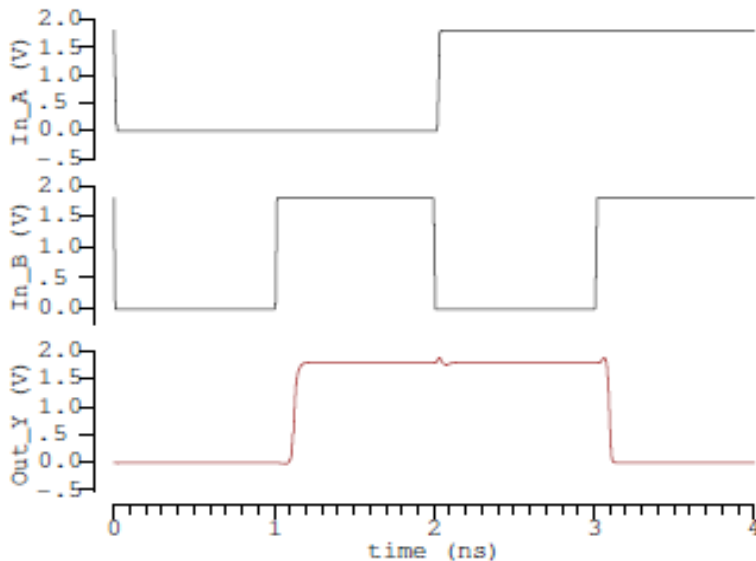
Εικόνα 36: Σχηματικό διάγραμμα μίας NAND πύλης με τη χρήση υβριδικής Memristor-CMOS λογικής.



Εικόνα 37: Σχηματικό διάγραμμα μίας XOR πύλης με τη χρήση υβριδικής Memristor-CMOS λογικής.

Η παροδική απόκριση της σχεδιασμένης XOR πύλης απεικονίζεται στην εικόνα 38. Η καθυστέρηση, ο χρόνος ανόδου και ο χρόνος καθόδου είναι 1.0 ps. Ένα σήμα παλμού με πλάτος 1.0 ns δίνεται ως είσοδος. Ο σχεδιασμός μιας XOR ή XNOR πύλης οδηγεί σε λογική υποβάθμιση όπως γίνεται αντιληπτό από την εικόνα 38. Ο λόγος για αυτήν την λογική υποβάθμιση είναι το κύκλωμα διαιρέτη τάσης. Η σχεδίαση μιας XOR ή XNOR πύλης χρειάζεται να γίνει σε κλιμακωτά στάδια όπως μπορούμε να δούμε και στην εικόνα 38, σε αυτή την περίπτωση μπορεί να χρησιμοποιηθεί ένας BUFFER για να επαναφέρουμε το επίπεδο των τάσεων εξόδου. Η έξοδος της XOR ή XNOR πύλης με τη χρήση BUFFER απεικονίζεται

στην εικόνα 38. Η τάση μετά την προσθήκη του BUFFER επανέρχεται στο επίπεδο του  $V_{CC} = 1.8V$ .



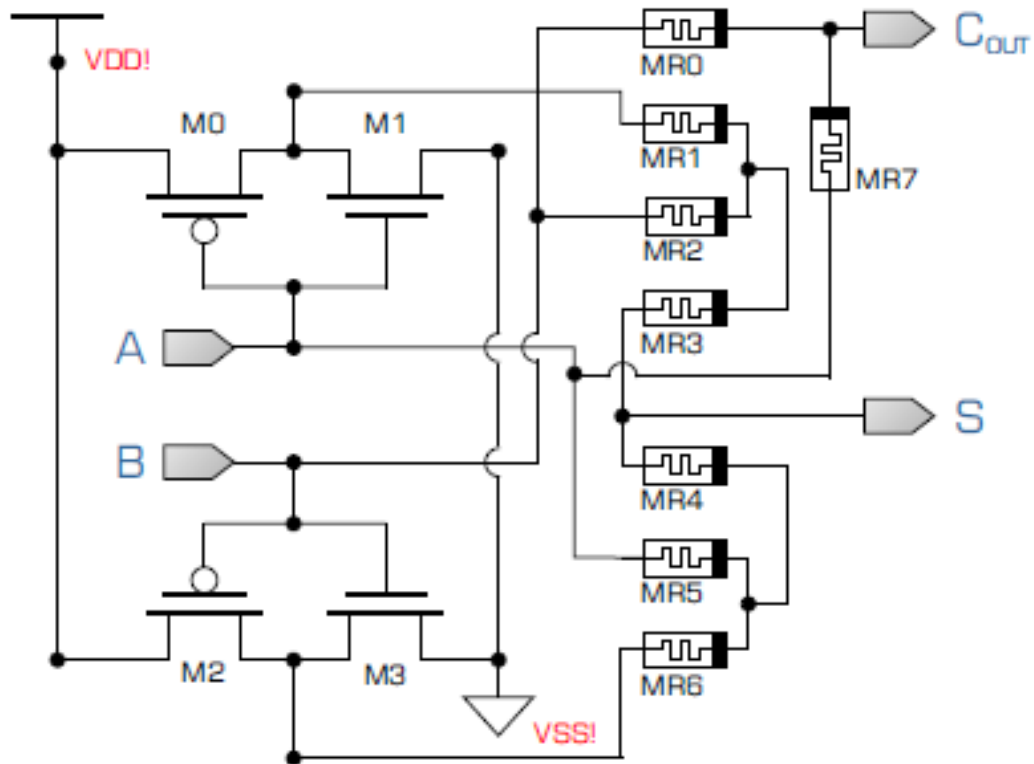
Εικόνα 38: Η παροδική απόκριση μιας XOR πύλης.

Με τη χρήση MeMOS λογικών πυλών, οποιοδήποτε ψηφιακό λογικό κύκλωμα μπορεί να δημιουργηθεί. Μετά από λεπτομερή καταγραφή και αξιολόγηση των παροδικών αποκρίσεων αυτών των πυλών οι οποίες χρησιμοποιούν CMOS λογική έχουμε φτάσει στο συμπέρασμα ότι οι πύλες οι οποίες έχουν σχεδιαστεί με MeMOS λογική επιδεικνύουν βελτιωμένες επιδόσεις.

### 3.3.3 Κυκλώματα Αθροιστών

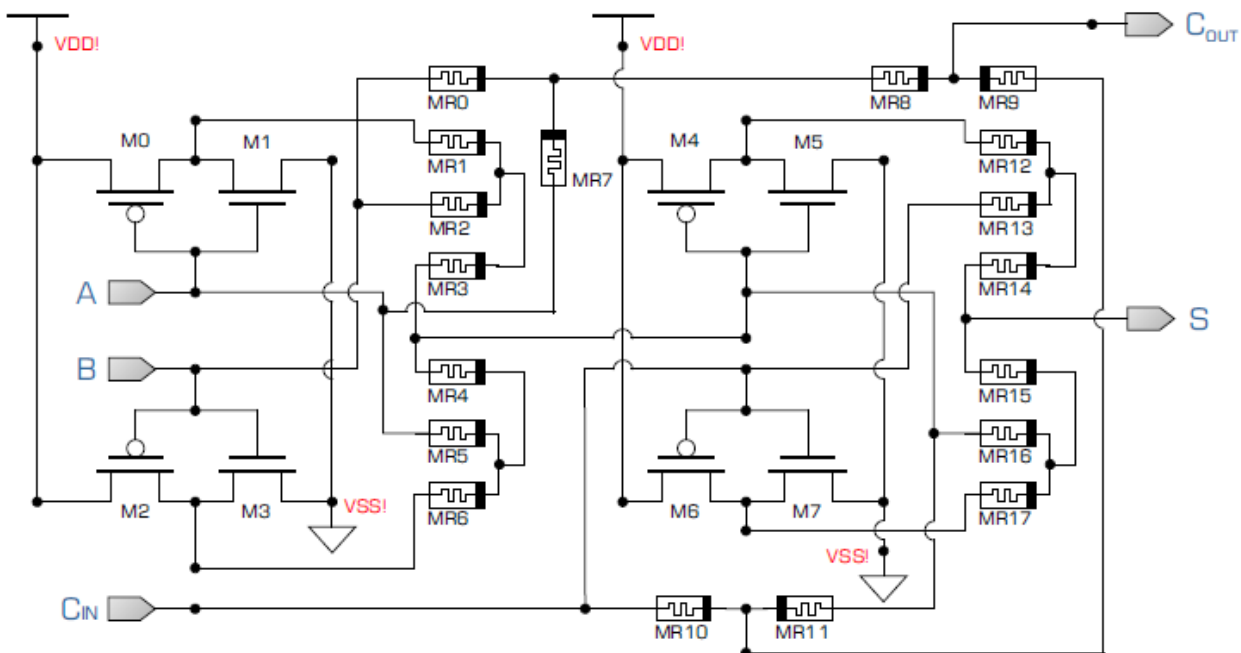
Χρησιμοποιώντας τις πύλες οι οποίες σχεδιάστηκαν με τη MeMOS λογική, μπορούμε να επεκτείνουμε περαιτέρω τα κυκλώματα προς την κατεύθυνση των βασικών δομικών μπλοκ οποιασδήποτε λογικής όπως για παράδειγμα της αθροιστικής λογικής. Οι αθροιστές χρησιμοποιούνται σε διαφορετικούς σχηματισμούς για να εκτελέσουν πρόσθεση, αφαίρεση, πολλαπλασιασμό ή διαίρεση bits. Η εικόνα 39 μας δείχνει ότι προσθέτοντας μόνο δύο memristors στο κύκλωμα της XOR πύλης, ένας ημιαθροιστής μπορεί να δημιουργηθεί.



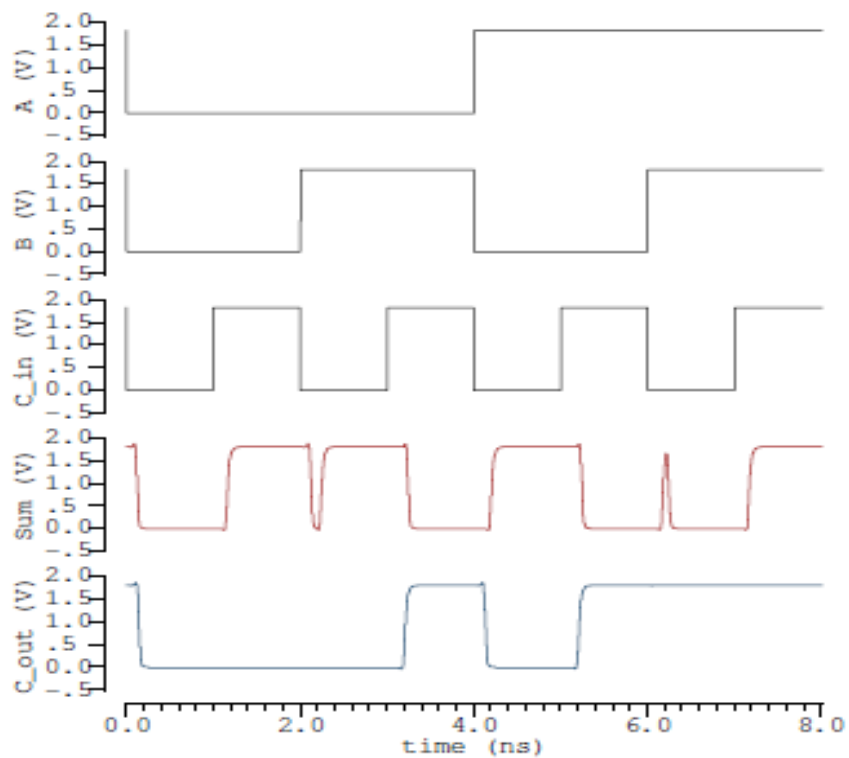


**Εικόνα 39:** Κύκλωμα ημιαθροιστή με τη χρήση υβριδικής Memristor-CMOS λογικής. Το κύκλωμα καταλαμβάνει την ίδια περιοχή με μία XOR πύλη με την ενσωμάτωση δύο μόνο επιπλέον memristors.

Στην εικόνα 37, τα memristor MR0 και MR1 συμπληρώνουν την δημιουργία μίας AND πύλης, η οποία είναι ίδια με αυτή των memristors MR4 και MR5, ενώ τα memristors MR2 και MR3 δημιουργούν μια πύλη OR. Η παραπάνω συνδεσμολογία τελικά οδηγεί στην κατασκευή μια XOR πύλης. Από την XOR πύλη, προσθέτοντας μία ακόμα AND πύλη στο τέλος της μπορούμε να δημιουργήσουμε έναν ημιαθροιστή όπως αυτός απεικονίζεται στην εικόνα 39. Σε αυτήν την περίπτωση τα memristors MR0 και MR7 συγκροτούν μια AND πύλη στο τέλος της XOR πύλης για να ολοκληρώσουν την κατασκευή του ημιαθροιστή. Παρομοίως, ένα κύκλωμα πλήρους αθροιστή ενός bit το οποίο έχει σχεδιαστεί χρησιμοποιώντας MeMOS λογική απεικονίζεται στην εικόνα 41 και αποτελείται από δύο πανομοιότυπους ημιαθροιστές και τα memristors MR0 και MR11 δρουν ως OR πύλη στο κύκλωμα έτσι ώστε να υλοποιήσουν μια λειτουργία CARRY. Η παροδική απόκριση του κυκλώματος του αθροιστή απεικονίζεται στην εικόνα 41.



Εικόνα 40: Εφαρμογή ενός κυκλώματος πλήρους αθροιστή το οποίο χρησιμοποιεί υβριδική Memristor-CMOS λογική. Το κύκλωμα αποτελείται από 8 MOSFETS. Ένας BUFFER απαιτείται στις εξόδους για να διορθώσει το πρόβλημα της λογικής υποβάθμισης.



Εικόνα 41: Παροδική απόκριση ενός κυκλώματος πλήρους αθροιστή για όλους τους πιθανούς συνδυασμούς.

### 3.3.4 Πλεονεκτήματα ως προς την λογική συμπερασμού

Ο υλικός συμπερασμός απέκτησε πρόσφατα μεγάλη δημοσιότητα όταν χρησιμοποίησε IMPLY βασισμένη σε memristors για να εκτελέσει όλες τις λειτουργίες [23]. Παρά το γεγονός ότι τα κυκλώματα που έχουν σχεδιαστεί για να χρησιμοποιούν IMPLY πύλες εμφανίζουν σημαντικά πλεονεκτήματα όπως η εκτέλεση λειτουργιών σε υψηλές ταχύτητες, καταλαμβάνουν μικρότερη έκταση και καταναλώνουν σημαντικά μικρότερα ποσά ενέργειας, η λειτουργία IMPLY λειτουργεί σε αντίσταση εσωτερικής κατάστασης των memristors. Εξωτερικά read/write κυκλώματα απαιτούνται για να ενσωματώσουμε την IMPLY βασισμένη λογική στην τρέχουσα γενιά CMOS. Τα κυκλώματα read/write καταλαμβάνουν πολύ περισσότερο χώρο εξαιτίας της πλήρους CMOS εφαρμογής. Επίσης λόγω των επιπλέον κυκλωμάτων οδηγούμαστε σε μία υποβάθμιση των επιδόσεων λόγω του bottlenecks της CMOS τεχνολογίας.

Η MeMOS λογική χρησιμοποιεί ένα υβρίδιο και των δύο τεχνολογιών. Το μεγαλύτερο πλεονέκτημα της χρήσης MeMOS λογικής είναι η ενοποίηση κυκλωμάτων με κυκλώματα τα οποία έχουν σχεδιαστεί μόνο με τη CMOS λογική. Με αυτόν τον τρόπο τα σήματα εισόδου και εξόδου λειτουργούν στο ίδιο επίπεδο τάσης το οποίο είναι αποδεκτό από τη CMOS λογική, και ως αποτέλεσμα η ανάγκη για επιπλέον κυκλώματα μειώνεται σημαντικά. Αν και καταλαμβάνουν περισσότερο χώρο από τα κυκλώματα τα οποία βασίζονται στην IMPLY λογική, η MeMOS λογική έχει πολλά πλεονεκτήματα ως προς την CMOS λογική και πλησιάζει σε πολλούς τομείς και την IMPLY.

### 3.3.5 Ανάλυση Επιδόσεων

Η παροδική απόκριση των αθροιστών οι οποίοι βασίζονται στη MeMOS λογική υπολογίζεται από διάφορες παραμέτρους όπως ο χρόνος ανόδου, ο χρόνος καθόδου και η καθυστέρηση. Οι παράμετροι αυτοί αναλύονται και συγκρίνονται με την τρέχουσα γενιά CMOS τεχνολογία. Ο πίνακας 11 δείχνει την ανάλυση επίδοσης διάφορων λογικών πυλών οι οποίες χρησιμοποιούν MeMOS λογική και ο πίνακας 12 δείχνει την ανάλυση επίδοσης πυλών οι οποίες χρησιμοποιούν CMOS λογική. Η επίδοση αναλύεται για ένα κύκλωμα πλήρους αθροιστή όπως φαίνεται και στην εικόνα 41. Η παροδική ανάλυση προσομοιώνεται για όλους τους πιθανούς συνδυασμούς του κυκλώματος. Το μεγαλύτερο πρόβλημα με το μοντέλο γραμμικής μετατόπισης ιόντων είναι οι μη σταθεροποιημένοι παράμετροι εξόδου το οποίο είναι γνωστό ως επίπεδο υποβάθμισης. Το επίπεδο υποβάθμισης μπορεί να προσεγγισθεί χάρη στο κύκλωμα του διαιρέτη τάσης και με αυτόν τον τρόπο το μοντέλο TEAM προτιμάται να αυξάνει το επίπεδο εξόδου του σήματος.

Πίνακας 11: Ανάλυση επίδοσης λογικών πυλών και κυκλωμάτων αθροιστή τα οποία χρησιμοποιούν υβριδική MEMRISTOR-CMOS λογική.

Logic Component	Rise Time ( $T_r$ )	Fall Time ( $T_f$ )	Delay $d$	Dyn+Sta Power
NOT	23.3 ps	14.1 ps	18.70 ps	0.5 $\mu$
AND	02.2 ps	00.8 ps	1.50 ps	1.50 $\mu$
OR	02.1 ps	00.8 ps	1.45 ps	1.51 $\mu$
NAND	23.4 ps	19.1 ps	21.25 ps	1.82 $\mu$
NOR	28.1 ps	14.2 ps	21.15 ps	1.83 $\mu$
XOR	40.4 ps	20.8 ps	30.60 ps	2.08 $\mu$
XNOR	40.1 ps	22.1 ps	31.11 ps	2.41 $\mu$
Half Adder	43.7 ps	22.4 ps	98.05 ps	8.07 $\mu$
Full Adder	82.1 ps	34.1 ps	212.3 ps	17.87 $\mu$
8bit FA	114.2 ps	78.7 ps	371.3 ps	52.71 $\mu$

Πίνακας 12: Ανάλυση επίδοσης λογικών πυλών και κυκλωμάτων αθροιστή τα οποία χρησιμοποιούν CMOS λογική.

Logic Component	Rise Time ( $T_r$ )	Fall Time ( $T_f$ )	Delay $d$	Norm. Power
NOT	23.3 ps	14.1 ps	18.70 ps	5.41 $\mu$
AND	35.0 ps	18.7 ps	26.85 ps	19.28 $\mu$
OR	29.4 ps	27.6 ps	28.51 ps	19.63 $\mu$
NAND	47.8 ps	20.9 ps	34.35 ps	10.69 $\mu$
NOR	50.7 ps	17.2 ps	33.90 ps	10.88 $\mu$
XOR	83.9 ps	48.3 ps	66.02 ps	47.81 $\mu$
XNOR	78.8 ps	50.4 ps	64.61 ps	43.61 $\mu$
Half Adder	85.2 ps	47.8 ps	126.2 ps	58.32 $\mu$
Full Adder	96.4 ps	54.2 ps	342.7 ps	117.3 $\mu$
8bit FA	183.1 ps	106.5 ps	586.2 ps	0.98 m

Στην παροδική ανάλυση του πλήρους αθροιστή υπάρχουν κάποιες μικρές δυσλειτουργίες για τις οποίες υπεύθυνη είναι η χρήση της CMOS τεχνολογίας. Τα memristors από μόνα τους παρέχουν μία σχεδόν ιδανική παροδική απόκριση για το κύκλωμα.

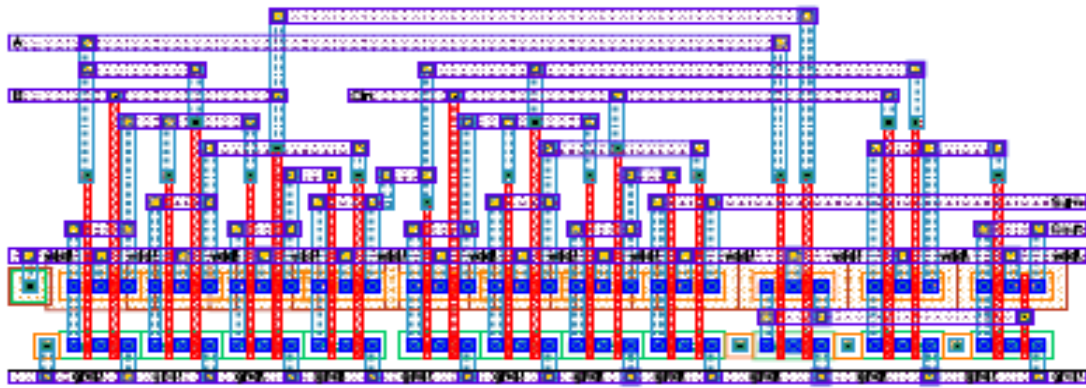
Η καθυστέρηση και ο χρόνος ανόδου και καθόδου των κυκλωμάτων για την δοθείσα παροδική απόκριση είναι  $T_r = 43.71$  ps,  $T_f = 22.43$  ps. Η καθυστέρηση η οποία υπολογίστηκε είναι 98 ps για το κύκλωμα του ημιαθροιστή. Η κανονικοποιημένη ισχύς για τον ημιαθροιστή για όλους τους πιθανούς συνδυασμούς είναι 8.07  $\mu$ W. Για το κύκλωμα πλήρους αθροιστή έχουμε μετρήσεις  $T_r = 82.12$  ps και  $T_f = 34.1$  ps και η καθυστέρηση η οποία αναλύθηκε είναι 213.3 ps. Η κανονικοποιημένη ισχύς για το κύκλωμα πλήρους αθροιστή για όλους τους πιθανούς συνδυασμούς του είναι 17.87  $\mu$ W.

Ο αθροιστής τεσσάρων bits σχεδιάζεται χρησιμοποιώντας τέσσερις πλήρης αθροιστές του ενός bit και μετά δύο αθροιστές των τεσσάρων bits συνδυάζονται για να δημιουργήσουν έναν αθροιστή των οχτώ bits. Οι παράμετροι επίδοσης για έναν πλήρη αθροιστή 8 bit είναι  $T_r = 114.2$  ps,  $T_f = 78.7$  ps, η καθυστέρηση είναι 371.3 ps για την χειρότερη περίπτωση και η κανονικοποιημένη ισχύς είναι 52.7  $\mu$ W.

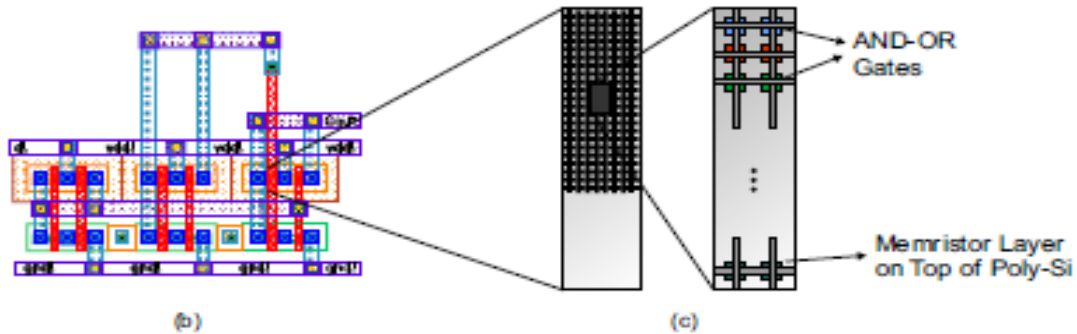
Για την υβριδική Memristor-CMOS λογική χρησιμοποιείται CMOS τεχνολογία επεξεργασίας των 180 nm. Προς χάριν της σύγκρισης, η πύλη AND χρειάστηκε 6 MOSFETs, ενώ στη περίπτωση της υβριδικής memristor-CMOS λογικής δεν χρειάστηκε κανένα MOSFET. Κατά αυτόν τον τρόπο ακόμα και για πύλες AND και OR n-εισόδων, υπάρχει μεγάλη εξοικονόμηση χώρου καθώς τα memristors μπορούν να αναπτυχθούν πάνω από τα MOSFETs και μέσω διασυνδέσεων οι πύλες μπορούν συνδεθούν μεταξύ τους.

Η διάταξη του CMOS πλήρους αθροιστή φαίνεται στην εικόνα 42(α). Στην εικόνα 42(b) απεικονίζεται μία πιθανή διάταξη του υβριδικού κυκλώματος memristor-CMOS λογικής με το στρώμα των memristor να είναι πάνω στο στρώμα του CMOS.

Τα memristor είναι συνδεδεμένα σε ένα στρώμα πολυπυριπίου στην κορυφή της MOSFET πύλης. Δύο memristor απαιτούνται για τον υπολογισμό της AND και OR λειτουργίας όπως απεικονίζεται και στην εικόνα 42(c). Πολλές πολύπλοκες λειτουργίες μπορούν να δημιουργηθούν οι οποίες θα καταλαμβάνουν την ίδια περιοχή με ένα MOSFET. Έτσι μπορεί να επιτευχθεί σημαντική μείωση στον χώρο και να εκτελεσθούν περισσότερες λογικές λειτουργίες ανά μονάδα χώρου του chip.



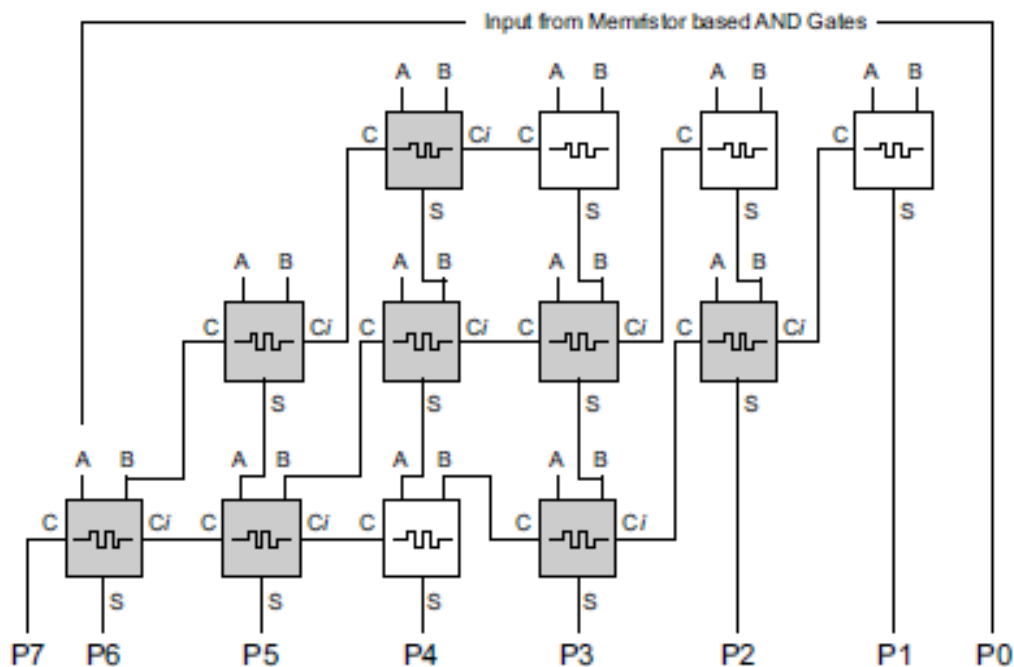
(a)



(b)

(c)

Εικόνα 42: (a) Διάταξη ενός κυκλώματος πλήρους αθροιστή με CMOS λογική το οποίο χρησιμοποιεί μια προσέγγιση με 36 transistor. (b). Διάταξη της CMOS στρώσης εάν χρησιμοποιείται σε μία εφαρμογή υβριδικής memristor-CMOS λογικής. Η περιοχή που καταλαμβάνει η CMOS στρώση μπορεί να μειωθεί κατά προσέγγιση 4 φορές. (c). Πιθανό σχέδιο διάταξης της στρώσης των memristor πάνω από την στρώση πολυπυριτίου των MOSFET. Πολλές πύλες AND και OR μπορούν να δημιουργηθούν μόνο στην στρώση πολυπυριτίου ενός MOSFET.



Εικόνα 43: Προτεινόμενη αρχιτεκτονική πολλαπλασιαστή με τη χρήση υβριδικής memristor-CMOS λογικής. Όλοι οι αθροιστές και οι πύλες έχουν δημιουργηθεί με τη χρήση MeMOS λογικής.

Οι παράμετροι απόδοσης των διαφόρων πυλών και των κυκλωμάτων αθροιστή δίνονται στον πίνακα 11 για την περίπτωση της υβριδικής memristor-CMOS λογικής και για την περίπτωση της CMOS λογικής δίνονται στον πίνακα 12. Ο αριθμός των transistors τα οποία απαιτούνται και στις δύο λογικές οικογένειες δίνονται στους πίνακες 13 και 14. Και τέλος η σύγκριση των παραμέτρων των κυκλωμάτων αθροιστή εμφανίζονται στον πίνακα 15.

Πίνακας 13: Αριθμός των MODFET και των memristors για λογικές πύλες με τη χρήση υβριδικής Memristor-CMOS και CMOS λογικής.

Device	NOT	AND	OR	NAND	NOR	XOR	XNOR	BUF
Hybrid Memristor-CMOS Logic								
MOSFETs	2	0	0	2	2	4	4	4
Memristors	0	2	2	2	2	6	6	0
CMOS Logic								
MOSFETs	2	6	6	4	4	12	12	4

**Πίνακας 14: Πλήθος MOSFET και Memristor για κύκλωμα αθροιστή το οποίο χρησιμοποιεί υβριδική CMOS-memristor λογική και CMOS λογική.**

Device	Half Adder	Full Adder	8-bit Adder
<b>Hybrid Memristor-CMOS Logic</b>			
MOSFETs	8	16	128
Memristors	8	18	144
<b>CMOS to Memristor Layer Transitions</b>			
VIAs	5	10	80
<b>CMOS Logic</b>			
MOSFETs	14	34	272

**Πίνακας 15: Σύγκριση παραμέτρων της υβριδικής CMOS-memristor λογικής με τη CMOS λογική.**

Parameter	Half Adder	Full Adder	8-bit Adder
Area Utilisation	57.2%	47%	47%
Included BUFFERS		50% of total area	
Performance		2.2x CMOS	
Complexity		Greater than CMOS	

### 3.3.6 Σύγκριση μεταξύ των λογικών οικογενειών

**Ταχύτητα:** Σε γενικές γραμμές ο ολικός χρόνος επεξεργασίας για οποιοδήποτε λογικό υπολογισμό είναι υβριδική memristor-CMOS λογική < CMOS λογική. Για παραμέτρους όπως χρόνος ανόδου, χρόνος καθόδου και καθυστέρηση η υβριδική memristor-CMOS λογική φαίνεται να έχει το πλεονέκτημα έναντι της CMOS λογικής. Η πύλη NOT χρησιμοποιείται για τη NOT λειτουργία και στις δύο λογικές οικογένειες, η οριοθέτηση της μέγιστης ταχύτητας της υβριδικής memristor-CMOS λογικής γίνεται καθαρά και μόνο λόγω της NOT πύλης.

**Εκμετάλλευση χώρου:** Με την υπόθεση ότι από τα MOSFETs τότε η εκμετάλλευση χώρου είναι υβριδική memristor-CMOS λογική < CMOS λογική. Αυτή η νέα λογική έχει επιφέρει θεαματικά αποτελέσματα στον τομέα της εξοικονόμησης χώρου καθώς τα memristor έχουν πλάτος 3 nm τα οποία είναι κατά πολύ μικρότερα από το πλάτος των 180 nm που έχει ένα MOSFET. Τα memristor μπορούν επίσης να τοποθετηθούν στην στρώση πολυπυριτίου ενός MOSFET και με αυτό τον τρόπο ένα MOSFET μπορεί να στεγάσει πολλά memristors. Πολλές περίπλοκες λειτουργίες μπορούν να επιτευχθούν στην περιοχή ενός μόνο MOSFET την ίδια στιγμή που στην CMOS λογική πολύ περισσότερα transistors απαιτούνται για την εκτέλεση της ίδιας λειτουργίας. Από τις παραπάνω εικόνες είναι εμφανές ότι υπάρχει μια μείωση της



τάξης του 47% στην περιοχή που καταλαμβάνεται αν γίνει χρήση της υβριδικής memristor-CMOS λογικής.

Πολυπλοκότητα χειρισμού: Η μόνη λογική οικογένεια η οποία χρειάζεται έναν εξωτερικό χειριστή διαβάσματος/εγγραφής είναι η IMPLY λογική οικογένεια. Επειδή η IMPLY λογική οικογένεια απαιτεί εσωτερικές καταστάσεις για λειτουργία διαβάσματος/εγγραφής, υπάρχει η ανάγκη μετατροπής αυτών εσωτερικών καταστάσεων από αντιστάσεις στις αντίστοιχες λογικές καταστάσεις για να μπορεί να συνεργαστεί με CMOS. Έτσι απαιτούνται χειριστές, κάτι το οποίο καταλαμβάνει πολύ χώρο και μπορεί να μειώσει περεταίρω την ταχύτητα της IMPLY λογικής.

Για την Υβριδική memristor-CMOS λογική και την CMOS λογική δεν απαιτούνται επιπλέον κυκλώματα για την εγγραφή και το διάβασμα. Ο πρωταρχικός ρόλος αυτής της υβριδικής memristor-CMOS λογικής είναι να κατασκευάσει μία νέα λογική οικογένεια η οποία να είναι σε θέση να συνεργαστεί με την τρέχουσας γενιάς CMOS τεχνολογίας.

Κατανάλωση ενέργειας: Γενικά η τρέχουσα τεχνολογία CMOS καταναλώνει περισσότερη ενέργεια εξαιτίας της συνεχούς σύνδεσης παροχής τάσης η οποία προκαλεί στατική διαρροή. Στην υβριδική memristor-CMOS λογική η στατική διαρροή είναι ακόμη εκεί. Παρόλα αυτά όμως η περίπτωση της είναι διαφορετική σε σχέση με την IMPLY λογική καθώς σε αυτή οι χειριστές οι οποίοι εμπεριέχει έχουν MOSFET πράγμα που σημαίνει ότι εμπεριέχει και κυκλώματα διαβάσματος/εγγραφής και άρα προκαλείται στατική κατανάλωση ισχύος [24].

Η στατική κατανάλωση ισχύος μπορεί να μειωθεί περαιτέρω με γνωστές τεχνικές όπως η περίφραξη ρολογιού. Η στατική κατανάλωση ισχύος δεν είναι ένα μείζων ζήτημα για την μέση ισχύ, η μέγιστη κατανάλωση ισχύος είναι η δυναμική ισχύς, η οποία οφείλεται στις διασυνδέσεις. Η ισχύς υπολογίζεται για σήματα τα οποία έχουν μέγιστες διασυνδέσεις. Η ισχύς κανονικοποιείται για τα memristors καθώς ο υπολογισμός της ισχύος δεν είναι ίδιος με τα CMOS [25].

Ευφυΐα: Μέχρι τώρα η CMOS λογική είναι η πιο ευφυής λογική, καθώς προσφέρει πολλούς διαφορετικούς τρόπους σχεδίασης ενός κυκλώματος ή μίας αρχιτεκτονικής σύμφωνα με τις ανάγκες ενός ολοκληρωμένου συστήματος. Όμως με την υβριδική memristor-CMOS λογική, η ευφυΐα αυξάνεται γιατί η στρώση των memristors βρίσκεται στην κορυφή της στρώσης του CMOS, πράγμα το οποίο δίνει ένα πλεονέκτημα στον υπολογισμό μερικών λειτουργιών σε ένα ξεχωριστό στρώμα και τα αποτελέσματα να μπορούν να παρθούν απευθείας από ένα άλλο στρώμα.

Επιπλέον βελτιώσεις: Στην υβριδική memristor-CMOS λογική, η ταχύτητα εξαρτάται σε μεγάλο βαθμό από τις τάσεις λειτουργίας οι οποίες καθορίζονται σε μεγάλο βαθμό από τις τάσεις κατωφλίου. Έτσι τα memristor με υψηλότερη τιμή ρεύματος κατωφλίου είναι γρηγορότερα αλλά καταναλώνουν περισσότερη ισχύ. Όσο για την εξοικονόμηση χώρου, αν η ταχύτητα είναι πρωτεύουσας σημασίας, το ρεύμα κατωφλίου πρέπει να υψηλό. Καθώς υπάρχει η ανάγκη για περισσότερους BUFFERS η οποία έχει αντίκτυπο στην εκμετάλλευση του χώρου.

Η ισχύς συνήθως καθορίζεται από τις εφαρμοζόμενες τάσεις, οι οποίες μπορούν να επιλεχθούν από το ρεύμα κατωφλίου. Μία μέθοδος για την μείωση της κατανάλωσης ενέργειας είναι η προσθήκη περισσότερων BUFFERS με CMOS λογικές πύλες. Οι BUFFERS

μπορούν να μειώσουν στην στατική κατανάλωση ισχύος καθώς και τη δυναμική κατανάλωση ισχύος καθώς οι BUFFERS μπορούν να εξαλείψουν εντελώς τις δυσλειτουργίες. Ωστόσο ο συμβιβασμός ως προς τον χώρο που αυτοί θα καταλαμβάνουν υφίσταται ακόμα. Η στατική ισχύς μπορεί να εξαλειφθεί εντελώς με την προσθήκη BUFFERS αμέσως μετά από κάθε διαδοχική υβριδική memristor-CMOS λογική κατάσταση.

## 4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

### 4.1 Συμπεράσματα

Τα memristors προσθέτουν νέες δυνατότητες στη CMOS τεχνολογία και σχεδιάζεται να είναι διαθέσιμα στο ευρύ κοινό στο άμεσο μέλλον. Σε αυτήν την εργασία, ερευνήσαμε τις ιδιότητες και τις επιπτώσεις των δυνατοτήτων των VLSI συστημάτων στην περιοχή των memristor. Αυτός ο συνδυασμός μόνιμης αποθήκευσης, πυκνών, γρήγορων, και χαμηλής κατανάλωσης συσκευών οι οποίες χρησιμοποιούνται στα συστήματα αποθήκευσης αλλά και στην κορυφή των CMOS, είναι μία νέα τεχνολογία, η οποία αλλάζει τον τρόπο που οι υπολογιστές οργανώνονται σήμερα.

Σε αυτήν την εργασία, παρουσιάστηκαν διαφορετικές οικογένειες για την εκτέλεση υπολογισμών με memristors, στην ενσωμάτωση τους στα CMOS αλλά και για λογική μέσα στη μνήμη. Η ενσωμάτωση των memristors στη CMOS είναι πλεονεκτική για την αύξηση του αριθμού των λογικών πυλών στην ίδια περιοχή (αύξηση της λογικής πυκνότητας). Ακόμα και χωρίς την μείωση του όγκου των CMOS transistors. Αυτή η προσέγγιση έχει πολλά επιπλέον πλεονεκτήματα καθώς επεκτείνει τον νόμο του Moore καθώς αυξάνει τον αριθμό των λογικών πυλών περισσότερο από τον παραδοσιακό παράγοντα δύο.

Μία διαφορετική προσέγγιση, η οποία ερευνάτε σε αυτή την εργασία, είναι η χρήση memristive-only λογική για λογική μέσα στη μνήμη. Αυτή η προσέγγιση εμπεριέχει μη-von Neumann αρχιτεκτονικές του in-memory computing. Σε αντίθεση με παλαιότερα προτεινόμενες in-memory computing αρχιτεκτονικές, η χρήση της memristive μνήμης δεν απαιτεί επιπλέον κυκλώματα ή την αλλαγή του κελιού μνήμης ή τη δομή του array της μνήμης. Η λογική μέσα στη μνήμη βασισμένη στα memristors είναι για αυτό το λόγο μέσα στη μνήμη διαδικασία και όχι απλά ένας συνδυασμός υπολογιστικών μηχανών και κελιών μνήμης. Είναι ακόμα απαραίτητος ο καθορισμός της πλήρους αρχιτεκτονικής για memristive computing in-memory, συμπεριλαμβανομένου και του σετ οδηγιών, χειρισμού και έρευνας των πιο κατάλληλων εφαρμογών για να πλεονεκτική σε όρους απόδοσης και ενέργειας.

Τέλος πρέπει να σημειωθεί ότι οι προτεινόμενες εφαρμογές για memory intensive computing σε αυτή την εργασία είναι μόνο η κορυφή του παγόβουνου, και στο κοντινό μέλλον πολλές καινοτόμες εφαρμογές θα προταθούν, αλλάζοντας τη δομή και την αρχιτεκτονική των υπολογιστών και των VLSI συστημάτων.

### 4.2 Μελλοντικές επεκτάσεις

Υπάρχουν πολλές αρχιτεκτονικές μνήμης οι οποίες χρειάζεται να διερευνηθούν, να σχεδιαστούν και τέλος να αξιολογηθούν. Για παράδειγμα, οι καταχωρητές πολλών καταστάσεων μπορούν να χρησιμοποιηθούν για επιπλέον μικροαρχιτεκτονικές. Ακόμα υπάρχουν μνήμες οι οποίες είναι σε θέση να περιλαμβάνουν από κοινού von Neumann μηχανές αλλά και άλλες. Είναι επίσης δυνατό να χρησιμοποιηθούν memristive μνήμες για data flow επεξεργαστές, associative επεξεργαστές αλλά και για reconfigurable μηχανές.

Σε αυτή την εργασία, περιγράφονται τρεις διαφορετικές λογικές οικογένειες για λογική μέσα σε

μια memristive μνήμη. Διάφορες Boolean λειτουργίες εκτελούνται με αυτές τις λογικές οικογένειες. Ωστόσο ο έλεγχος αυτών λογικών οικογενειών δεν εξερευνήθηκε. Είναι απαραίτητη η ανάπτυξη μίας ολοκληρωμένης αρχιτεκτονικής για computing μέσα στη μνήμη, με τη χρήση βασικών memristive λογικών πυλών. Επίσης πρέπει να αναπτυχθούν αλγόριθμοι για την εκτέλεση οποιασδήποτε λειτουργίας Boolean, απαιτείται νέο σετ οδηγιών, ένας μεταγλωττιστής πρέπει να σχεδιαστεί, και τέλος πρέπει να καθοριστεί η δομή του συστήματος η οποία θα περιλαμβάνει μία διεπαφή ανάμεσα στη κεντρική μονάδα επεξεργασίας και τη memristive μνήμη.

**ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ**

<b>Ξενόγλωσσος όρος</b>	<b>Ελληνικός Όρος</b>
processing elements	στοιχεία επεξεργασίας
Semiconductor device	Ημιαγωγική διάταξη
the memory wall	το τείχος της μνήμης
error bit rate	ρυθμός λάθος bit
resistive random access memory	ωμική μνήμη τυχαίας προσπέλασης
circuit element	στοιχείο κυκλώματος
polarity	Πόλικότητα
Titanium dioxide	διοξειδίου του Τιτανίου
static noise margin	περιθώριο στατικού θορύβου
phase change memory	μνήμη αλλαγής φάσης
clock cycle	κύκλος ρολογιού
writing times	χρόνοι εγγραφής
logic gate	λογική πύλη
state drift	ολίσθηση κατάστασης
memory cells	κελιά μνήμης
material implication	υλικός συμπερασμός
operator	Τελεστής
computational stages	υπολογιστικά στάδια
ratio	αναλογία
structure	δομή

## ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

CMOS	Complementary symmetry Metal Oxide Semiconductor
CFMT	Continuous Flow Multithreading
DRAM	Dynamic Random Access Memory
MRL	Memristor Ratioed Logic
PCM	Phase Change Memory
TEAM	Threshold Adaptive Memristor
VLSI	Very Large Scale Memristor
MAGIC	Memristor Aided Logic
SNM	Static Noise Margin
MCAM	Memristor based Content Addressable Memory
RRAM	Resistive Random Access Memory
SSD	Solid State Drive

## ΑΝΑΦΟΡΕΣ

- [1] S. E Thomson and S. Parthasarathy, "Moore's Law: the Future of Si Microelectronics," *Materials Today*, Vol 9, No.6, pp. 20-25, June 2006.
- [2] W.A Wulf and S. A. Mckee, "Hitting the Memory Wall: Implications of the Obvious," *ACM SIGARCH Computer Architecture News*, Vol 23, No. 1, pp.20-24, March 1995.
- [3] J. Scaramuzzo, "The Flash Transformed Data Center," *Proceedings of the Nonvolatile Workshop*, March 2014
- [4] J. Akerman, "Toward a Universal Memory," *Science*, Vol. 308, No.5721, pp.508-510, April 2005.
- [5] L. O. Chua, "Memristor – The Missing Circuit Element," *IEEE Transactions on Circuit Theory*, Vol 18, No. 5, pp. 507-519
- [6] L. O. Chua and S. M. Kang, "Memristive Devices and Systems," *Proceedings of the IEEE*, Vol 64, No 2, pp. 209-223, February 1976
- [7] N. Verma and A.P Chandrakasan. A 256 kb X 65 nm 8T subthreshold SRAM employing sense-amplifier redundancy. *IEEE J Solid-St Circ*, 43(1):141-149, Jan. 2008
- [8] G. Snider, "Computing with Hysteric Resistor Crossbars," *Applied Physics A: Materials Science and Processing*, Vol. 80, No.6 pp.1165-1172, March 2005.
- [9] S. Kvatinsky, E.G. Friedman, A. Kolodny, and U. C. Weiser, "Memristor-based IMPLY Logic Design Procedure," *Proceedings of the IEEE International Conference on Computer Design*, pp. 142-147, October 2011
- [10] E. Lehtonen and M.Laiho, "Stateful Implication Logic with Memristors," *Proceedings of the IEEE/ACM International Symposium on Nanoscale Architectures*, pp33-36, July 2009.
- [11] E. Lehtonen, J. H. Poikonen, and M. Laiho, "Two Memristors Suffice to Compute All Boolean Functions," *Electronics Letters*, Vol. 46, No.3, pp.239-240, February 2010.
- [12] A. Flocke and T. G. Noll, "Fundamental Analysis Of Resistive Nano-Crossbars for the Use in Hybrid Nano/CMOS-memory," *Proceedings of the Euripean Solid State Circuits Conference*, pp. 328-331, September 2007.
- [13] M. A. Zidan and K. N. Salama, "Memristor Based Memory: The Sneak Paths Problem and Solutions," *Microelectronics Journal*, 2012 (in press).
- [14] S. Kvatinsky, E. G.Friedman, A. Kolodny, and U. C. Weiser, "TEAM – ThrEshold Adaptive Memristor Model." *IEEE Transactions on Circuits and Systems: Regular Papers*, Vol 60, No. 1, pp.211-221, January 2013.
- [15] S. Kvantinsky, E. G.Friedman, A. Kolodny, and U. C. Weiser, "Memristor-based IMPLY Logic Design Procedure," *Proceedings of the IEEE International Conferernce on Computer Design*, pp.142-147, October 2011.
- [16] A.C. Torrezan, J.P. Strachan, G. Medeiros-Ribiero, and R. S. Williams, "Sub-Nanosecond Switching of a Tantalum Oxide Memristor," *Nanotechnology*, Vol. 22, No. 48, pp. 1-7, November 2011.
- [17] Y. V. Pershin and M. Di Ventra, "Practical Approach to Programmable Analog Circuits with Memristors," *IEEE Transactions on Circuits and Systems: Regular Papers*, Vol. 57 No. 8, pp.1857-1864, August 2010.
- [18] S. Kvantinsky, E. G. Friedman, A. Kolodny and U. C. Weiser, "Memrisotr-based Material Implication (IMPLY) Logic: Design Principles and Methodologies," *IEEE Transactions on Very Large Scale Integration (VLSI)* (in press).
- [19] S. Kvantinsky, M. Ramadan, E. G. Friedman, and A. Kolodny, "VTEAM – A General Model for Voltage Controlled Memristors," *CCIT Technical Report #856*, April 2014.
- [20] J. J. Yang, D. B. Strukov, and D. R. Stewart, "Memristive Devices for Computing," *Nature Nanotechnology*, Vol. 8, pp.13-24, January 2013.
- [21] E. Linn, R. Rosezin, C. Kugeler, and R. Waser, "Complementary Resistive Switches for Passive Nanocrossbar Memories," *Nature Materials*, Vol. 9, No. 5, pp. 403-406, April 2010.
- [22] J.Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Steward, and R. S. Williams, "Memristive switches enable stateful logic operations via material implication," *Nature*, Vol. 464, No. 7290, pp.873-876, 2010.
- [23] B. Hu, Y. Zhang, W. Chen, C. Xu, and Z. L. Wang, "Self-heating and external strain coupling induced phase transition of vo2 nanobeam as single domain switch," *Advanced Materials*, Vol. 23, No. 31, pp. 3536-3541, 2011.
- [24] T. Chan, J. Chen, P. Ko, and C. Hu, "The impact of gate-induced drain leakage current on mosfet scaling," in *Electron Devices Meeting, 1987 International*, Vol.33, IEEE, 1987, pp.718-721.
- [25] S.Shin, K. Kim, and S. Kang, "Recobfigurable stateful nor gate for large-scale logic-array integrations," *Circuits and Systems: Express Briefs, IEEE Transactionson*, Vol. 58, No 7, pp 442-446, 2011.